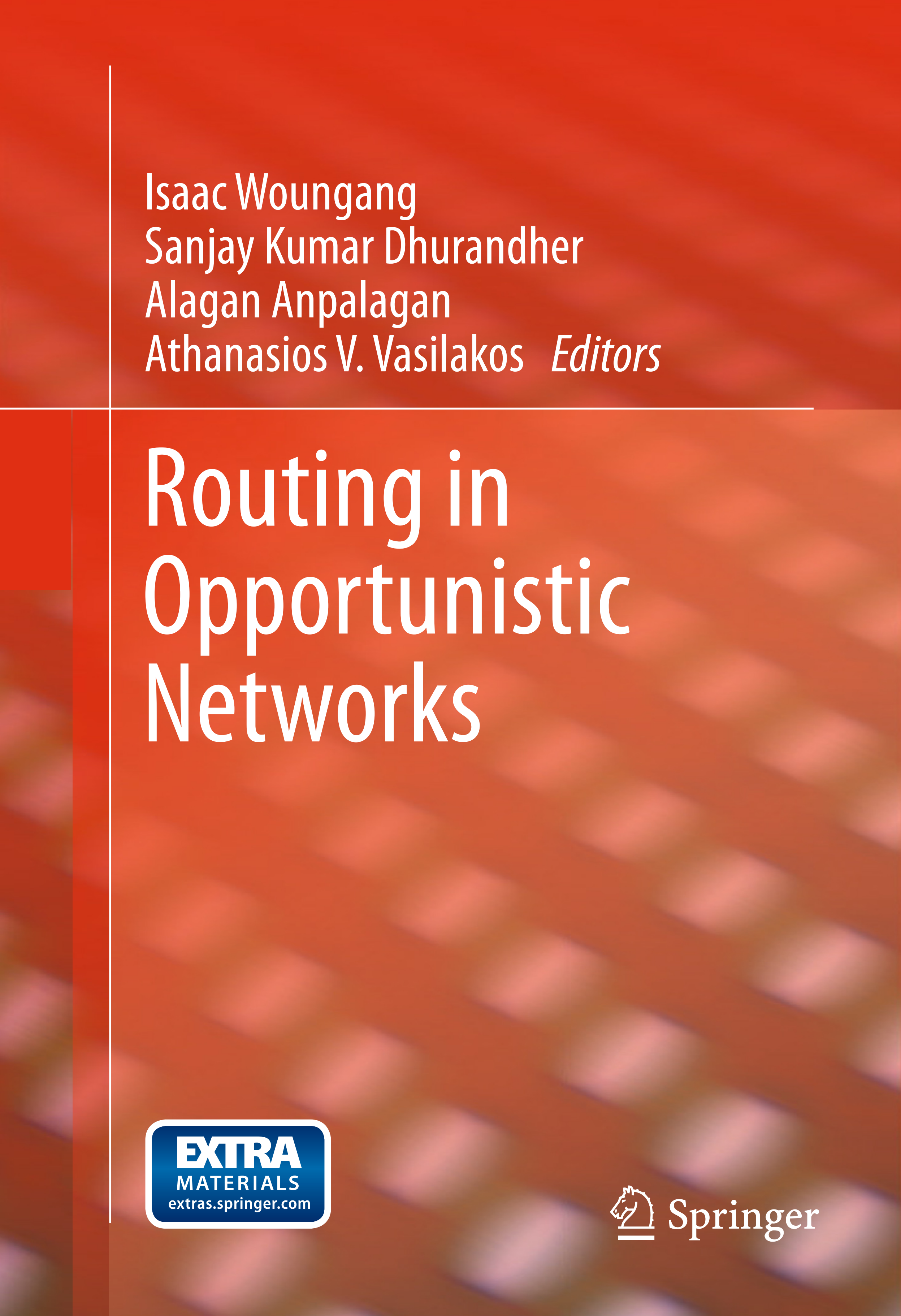


Isaac Woungang
Sanjay Kumar Dhurandher
Alagan Anpalagan
Athanasios V. Vasilakos *Editors*

Routing in Opportunistic Networks

EXTRA MATERIALS
extras.springer.com

Springer

Routing in Opportunistic Networks

Isaac Woungang · Sanjay Kumar Dhurandher
Alagan Anpalagan · Athanasios V. Vasilakos
Editors

Routing in Opportunistic Networks

Springer

Editors
Isaac Woungang
Department of Computer Science, George Vari Engineering and Computing Centre
Ryerson University
Toronto, ON
Canada

Sanjay Kumar Dhurandher
Division of Information Technology, Netaji Subhas Institute of Technology
University of Delhi
New Delhi
India

Alagan Anpalagan
Department of Electrical and Computer Engineering, George Vari Engineering and Computing Centre
Ryerson University
Toronto, ON
Canada

Athanasios V. Vasilakos
University of Western Macedonia
Nea Erythraia
Greece

Additional material to this book can be downloaded from http://extras.springer.com.

ISBN 978-1-4614-3513-6 ISBN 978-1-4614-3514-3 (eBook)
DOI 10.1007/978-1-4614-3514-3
Springer New York Heidelberg Dordrecht London

Library of Congress Control Number: 2013935479

© Springer Science+Business Media New York 2013
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Overview and Goals

Due to recent advances in wireless technologies such as 3G, Bluetooth, WiFi, to name a few, a pragmatic evolution of the generic mobile ad hoc networks paradigm (so-called opportunistic networks) has emerged. Opportunistic networks are a category of challenged mobile ad hoc networks different from legacy ad hoc networks. In opportunistic networks, unpredictable and unstable topologies, prolonged disconnections, and partitions can occur frequently, leading to a paradigm shift for the design of network services. In such environment, routing is considered as a challenging activity. To our knowledge, there have been very few reference books covering the topic of routing in opportunistic networks. This book aims to provide a comprehensive guide on new ideas and results on this topic. It is anticipated that the book will be a valuable reference for educationists, instructors, researchers, and practitioners in this field.

The book provides a comprehensive guide to selected topics, both ongoing and emerging, in the area of routing in opportunistic networks, using a treatment approach suitable for pedagogical purposes. Consisting of contributions from worldwide well-known and high profile researchers in their respective specialties, selected topics that are covered in this book are related to mobility and routing, social-aware routing, context-based routing, energy-aware routing, incentive-aware routing, stochastic routing, modeling of intermittent connectivity, in both infrastructure and infrastructure-less opportunistic networks.

The book has been prepared keeping in mind that it needs to prove itself to be a valuable resource, dealing with both the important core and the specialized issues in routing in opportunistic networks. We hope that it will be a valuable resource and reference for instructors, researchers, students, engineers, scientists, mangers, and industry practitioners in these fascinating fields. All chapters are integrated in a manner that renders the book as a supplementary reference volume and textbook for use in both undergraduate and graduate courses on routing in opportunistic networks. Each chapter is of an expository, but also of a scholarly or survey style, on a particular topic within the scope of routing in opportunistic networks.

Organization and Features

The book is organized into 14 chapters, each chapter is written by topical area experts.

Chapter 1 investigates mobility-enabled message dissemination approaches for opportunistic networks. A survey on mobility models, analytical results on motion characteristics, and mobility-based Delay Tolerant Network routing strategies, is presented, showing that analytical mobility properties and abstracted graphic features are significant to the success of some routing protocols.

Chapter 2 In the recent years, a new routing trend based on social similarity emerged where social relationships are used to improve opportunistic routing. This Chapter investigates the different approaches to opportunistic routing, with focus on social-aware approaches, and how they make use of social information derived from opportunistic contacts to improve the data forwarding. A state-of-the-art survey on existing taxonomies for opportunistic routing based on the new social trends, are provided along with a set of experiments in scenarios based on synthetic mobility models and human traces, demonstrating the potential of social-aware routing protocols for opportunistic networks.

Chapter 3 emphasizes the role that context information plays when forwarding decisions are taken in opportunistic networks. Representative context-based routing protocols for OppNets so far proposed in the literature are classified on the basis of context information of users that they exploit, leading to two main classes of routing protocols, namely context-oblivious and context-aware protocols. Context-aware routing protocols are further classified into two subclasses, referred to as partially context-aware and fully context-aware protocols, based on the amount of context information that they use for routing.

Chapter 4 proposes an in-depth analysis of opportunistic relaying efficiency under different realistic radio channel conditions, with focus on finding the optimal tradeoff between energy and latency minimizations as optimization objectives. An optimal bound, so-called the Pareto front of the related optimization problem, is derived, which offers a good insight into the advantages of opportunistic routing compared to classical multi-hop routing schemes.

Chapter 5 provides an analysis of the infrastructure needed for communication in Opportunistic Networks. Representative routing protocols for message passing in order to optimize message several key parameters such as delivery delays, energy consumption, buffer occupancy, bandwidth requirements and throughput, are described in-depth. Important and explicit characteristics of each studied routing protocols along with their areas of application, are also highlighted.

Chapter 6 provides an overview of routing protocols that exploit intermittent communication opportunities for increased data transmission success in mobile ad hoc networks. These protocols are further characterized with respect to three aspects, namely the manner in which each protocol discovers and exploits the intermittent communication opportunities in a mobile environment; the way that each protocol addresses multicast; and the type of real-world applications that are enabled by the use each of the studied opportunistic routing protocols.

Chapter 7 analyzes the connectivity issues in a particular type of opportunistic networks known as vehicular ad hoc networks. It is argued that vehicle-to-vehicle communications may not necessary be the most appropriate interconnection scheme for data delivery in sparse or totally disconnected scenarios, and vehicle-to-infrastructure communications may represent a viable solution to bridge the inherent network fragmentation that can exist in multi-hop networks formed over moving vehicles. In this regard, recent related work focusing on vehicular connectivity models, as well as review of hybrid and opportunistic vehicular communication paradigms designed to improve connectivity, are discussed in-depth.

Chapter 8 provides a detailed review of probabilistic schemes for routing in ad-hoc opportunistic networks, where a handful of most influential such schemes are selected and discussed. A recently proposed simulation framework whose goal is to provide an understanding of how lower layer conditions affect the decision of the probabilistic parameters in probabilistic schemes is also provided.

Chapter 9 examines various mathematical models for the performance evaluation of ad hoc opportunistic routing protocols. A novel analytical model is derived where all consecutive forwarders are in the transmission range of one another. As a proof of concept, the proposed analytical model is shown to minimize the packet retransmissions.

Chapter 10 provides a holistic view of research efforts toward the design and development of transport protocols for delay tolerant networks environments. A state-of-the-art survey of transport protocols for delay tolerant networks is presented. Based on this study, a novel protocol targeted at terrestrial delay tolerant network environments consisting of a large number of highly mobile nodes with random mobility is proposed.

Chapter 11 provides investigates four research areas of opportunistic routing, namely, routing metrics, candidate section, candidate coordination, geographic opportunistic routing, and multicast opportunistic routing. The main research contributions in each of these areas are surveyed, with focus on performance analysis studies as objective. A new DTMC model is introduced to analyze the performance gain that may be achieved when using opportunistic routing. Simulation results are provided to validate the effectiveness of the proposed model.

Chapter 12 introduces the utilizations of social characteristics such as community, centrality, similarity, and friendship, in the design of social-based routing protocols for opportunistic networks. Recent social-based opportunistic routing approaches which use these social characteristics to assist in packet forwarding are presented and qualitatively compared against those characteristics.

Chapter 13 provides a survey of representative routing protocols for infrastructure-less opportunistic networks. These routing protocols are analyzed and qualitatively compared on the basis of their advantages and disadvantages with respect to a variety of parameters, leading to the identification of some critical and explicit characteristics of each protocol along with their areas of application.

Chapter 14 focuses on the requirements for incentive mechanisms in opportunistic networks. Several effective incentive mechanisms for opportunistic networks are discussed, and challenges for future research in this area are investigated and grouped into the following categories: user behavior, using social network information, cross-layer information use, modelling social network behavior, and metrics for analyzing incentive mechanisms.

Below are some of the important features of this book, which, we believe, would make it a valuable resource for our readers:

- This book is designed, in structure and content, with the intention of making the book useful at all learning levels.
- Most of the chapters of the book are authored by prominent academicians, researchers, and practitioners, with solid experience and exposure to research on the area of routing in opportunistic networks. These contributors have been working with in this area for many years and have thorough understanding of the concepts and practical applications.
- The authors of this book are distributed in a large number of countries and most of them are affiliated with institutions of worldwide reputation. This gives this book an international flavor.
- The authors of each chapter have attempted to provide a comprehensive bibliography section, which should greatly help the readers interested further to dig into the aforementioned research area.
- Throughout the chapters of this book, most of the core research topics have been covered, making the book particularly useful for industry practitioners working directly with the practical aspects behind enabling the technologies in the field.

We have attempted to make the different chapters of the book look as much coherent and synchronized as possible. However, it cannot be denied that due to the fact chapters were written by different authors, it was not fully possible to achieve this task. We believe that this is a limitation of most edited books of this kind.

Target Audience

The book is written to primarily target the student community. This includes the students of both senior undergraduate and graduate levels—as well as students having an intermediate level of knowledge of the studied topics, and those having extensive knowledge about many of the topics. To keep up this goal, we have attempted to design the overall structure and content of the book in such a manner that makes it useful at all learning levels. The secondary audience for this book is the research community, in academia or in the industry. We have also taken into consideration the needs to those readers, typically from the industries, who have quest for getting insight into the practical significance of the topics, expecting to discover how the spectrum of knowledge and the ideas are relevant for real-life applications of the ‘opportunistic network’ paradigm and technologies.

Acknowledgments

We are extremely thankful to the 40 authors of the 14 chapters of this book, who have worked very hard to bring this unique resource forward for helping the students, researchers, and community practitioners. We feel that it is contextual to mention that as the individual chapters of this book are written by different authors, the responsibility of the contents of each of the chapters lies with the concerned authors.

We like to thank Mr. Brett Kurzman, Publishing Editor who worked with us on the project from the beginning, for his professionalism. We also thank Springer and Ms. Rebecca R. Hytowitz (Springer Editorial Assistant), who tirelessly worked with us and helped us in the publication process.

Isaac Woungang
Sanjay Kumar Dhurandher
Alagan Anpalagan
Athanasios V. Vasilakos

Contents

Contributors

Arshad Ali Telecom SudParis, 9 rue C. Fourier, Evry Cedex 91011, France, e-mail: arraza92@yahoo.com

Eitan Altman INRIA, 2004 Route des Lucioles, Sophia-Antipolis 06902, France, e-mail: Eitan.Altman@sophia.inria.fr

Vangelis Angelakis Department of Science and Technology, Linköping University, Luntgatan 2, Norrköping Campus, 60174 Norrköping, Sweden, e-mail: vangelis.angelakis@liu.se

Olivier Berder IRISA, INRIA, Université de Rennes 1, France, 6 rue de kerampont, Lannion 22300, France, e-mail: olivier.berder@irisa.fr

Shruti Bhati CAITFS, Division of Information Technology, Netaji Subas Institute of Technology, University of Delhi, Azad Hind Fauj Marg, Sector-3, Dwarka (Pappankalan) 110 078, New Delhi, India, e-mail: sb.shrutibhati@gmail.com

Greg Bigwood University of St Andrews, Fife, UK; Jack Cole Building, North Haugh, St Andrews KY169SX, UK, e-mail: gjb4@st-andrews.ac.uk

Claudia Campolo Department DIMET, University Mediterranea of Reggio Calabria, Località, Feo di Vito, 89126 Reggio Calabria, Italy, e-mail: claudia.campolo@unirc.it

Llorenç Cerdà-Alabern Departament d′Arquitectura de Computadors, Universitat Politecnica de Catalunya, Campus Nord, Mòdul C6, despatx 221, Jordi Girona, 1-3, 08034 Barcelona, Spain, e-mail: llorenc@ac.upc.edu

Tijani Chahed Telecom SudParis, 9 rue C. Fourier, Evry Cedex 91011, France, e-mail: Tijani.Chahed@it-sudparis.eu

Amir Darehshoorzadeh Robson Eduardo De Grande, 48 Gagnon street–Apartment #2, Gatineau, QC J8X 1Y4, Canada, e-mail: amir@ac.upc.edu

Sanjay Kumar Dhurandher CAITFS, Division of Information Technology, Netaji Subas Institute of Technology, University of Delhi, Azad Hind Fauj Marg, Sector-3, Dwarka (Pappankalan) 110 078, New Delhi, India, e-mail: dhurandher@gmail.com

Sourabh Goyal Division of Computer Engineering, Netaji Subas Institute of Technology, University of Delhi, Azad Hing Fauj Marg, Sector-3, Dwarka (Pappankalan) 110 078, New Delhi, India, e-mail: sourabh912@gmail.com

Bo Gu SMC Networks, Inc., 12800 Turtle Rock Rd, Apt 12206, Austin, TX 78729, USA, e-mail: bgu@cs.ua.edu

Sahil Gupta Division of Computer Engineering, Netaji Subas Institute of Technology, University of Delhi, Azad Hing Fauj Marg, Sector-3, Dwarka (Pappankalan) 110 078, New Delhi, India, e-mail: sahilgupta221@gmail.com

Tristan Henderson University of St Andrews, Fife, UK; Jack Cole Building, North Haugh, St Andrews KY169SX, UK, e-mail: tnhh@st-andrews.ac.uk

Xiaoyan Hong Department of Computer Science, The University of Alabama, 101 Houser Hall, 301 Seventh, Avenue, Tuscaloosa, AL 35401, USA, e-mail: hxy@cs.ua.edu

Aniket Ingavale Department of Computer Science, ABV-Indian Institute of Information Technology and Management, Morena Link Road, Gwalior 474010, Madhya Pradesh, India

Thomas D. C. Little Department of Electrical and Computer Engineering, Boston University, 8 Saint Mary's Street, Boston, MA 02215, USA, e-mail: tdcl@bu.edu

Arka Prokash Mazumdar Department of CSE, IIT Patna, Navin Government Polytechnic Campus, Patliputra Colony, Patna 800 013, India, e-mail: arka@iitp.ac.in

Paulo Mendes SITI, Universidade Lusófona, Campo Grande, 376, Ed. U, 1° Piso, 1749-024 Lisbon, Portugal, e-mail: paulo.mendes@ulusofona.pt

Pramita Mitra Department of Computer Science and Engineering, University of Notre Dame, 325D Cushing Hall, Notre Dame, IN 46556, USA, e-mail: pmitra@nd.edu

Antonella Molinaro Department DIMET, University Mediterranea of Reggio Calabria, Località, Feo di Vito, 89126 Reggio Calabria, Italy, e-mail: antonella.molinaro@unirc.it

Waldir Moreira SITI, Universidade Lusófona, Campo Grande, 376, Ed. U, 1° Piso, 1749-024 Lisbon, Portugal, e-mail: waldir.junior@ulusofona.pt

Manoj Panda Telecom SudParis, 9 rue C. Fourier, Evry Cedex 91011, France, e-mail: manoj.manojpanda@gmail.com

K. K. Pattanaik Department of Computer Science, ABV-Indian Institute of Information Technology and Management, Morena Link Road, Gwalior 474010, Madhya Pradesh, India

George Perantinos Forthnet S.A–Research and Development Department, Science and Technology Park of Crete STEP, Crete, PO Box 2219, Vassilika Vouton, Heraklion, Crete, Greece

Vicent Pla ETSIT (edif. 4D), Universitat Politècnica de València, Camino de Vera, s/n, 46022 Valencia, Spain, e-mail: vpla@dcom.upv.es

Christian Poellabauer Department of Computer Science and Engineering, University of Notre Dame, 325D Cushing Hall, Notre Dame, IN 46556, USA, e-mail: cpoellab@cse.nd.edu

Ashok Singh Sairam Department of CSE, IIT Patna, Navin Government Polytechnic Campus, Patliputra Colony, Patna 800 013, India, e-mail: ashok@iitp.ac.in

Lucile Sassatelli Laboratoire I3S, Université Nice Sophia Antipolis, CNRS, Nice, France, e-mail: sassatelli@i3s.unice.fr

Olivier Sentieys IRISA, INRIA, Université de Rennes 1, France, 6 rue de kerampont, Lannion 22300, France, e-mail: olivier.sentieys@irisa.fr

Deepak Kumar Sharma Division of Computer Engineering, Netaji Subas Institute of Technology, University of Delhi, Azad Hing Fauj Marg, Sector-3, Dwarka (Pappankalan) 110 078, New Delhi, India, e-mail: dk.sharma1982@yahoo.com

Elias Tragos Institute of Computer Science, Foundation for Research and Technology-Hellas (FORTH), N. Plastira 100, Vassilika Vouton, 700 13 Heraklion, Crete, Greece, e-mail: etragos@ics.forth.gr

Anna Maria Vegni Department of Applied Electronics, University of Roma Tre, Via della Vasca Navale, 84, 00146 Rome, Italy, e-mail: amvegni@uniroma3.it

Anshul Verma Department of Computer Science, ABV-Indian Institute of Information Technology and Management, Morena Link Road, Gwalior 474010, Madhya Pradesh, India, e-mail: anshulverma87@gmail.com

Yu Wang Department of Computer Science, University of North Carolina at Charlotte, 9201 University City Blvd, Charlotte, NC 28223, USA, e-mail: Yu.Wang@uncc.edu

Isaac Woungang Department of Computer Science, Ryerson University, 350 Victoria Street, Toronto, ON M5B 2K3, Canada, e-mail: iwoungan@scs.ryerson.ca

Di Yuan Department of Science and Technology, Linköping University, Luntgatan 2, Norrköping Campus, 60174 Norrköping, Sweden, e-mail: diyua@itn.liu.se

Ruifeng Zhang IRISA, INRIA, Université de Rennes 1, France, 6 rue de kerampont, Lannion 22300, France, e-mail: ruifeng.zhang@inria.fr

Ying Zhu Department of Computer Science, University of North Carolina at Charlotte, 9201 University City Blvd, Charlotte, NC 28223, USA, e-mail: yzhu17@uncc.edu

Chapter 1
Identifying the Intertwined Links Between Mobility and Routing in Opportunistic Networks

Xiaoyan Hong and Bo Gu

Abstract Mobility intertwines with routing protocols play a vital role in opportunistic networks. First and foremost, mobility creates opportunities for mobile nodes to connect and communicate when they encounter. A series of encounter opportunities can spread a message among many nodes and to a large area until eventual delivery. Further, mobility properties, when utilized by routing protocols, can greatly improve performance. This chapter will trace the research on mobility and mobility enabled message dissemination approaches, and will present a survey over mobility models, analytical results on motion characteristics, and routing strategies that largely rely on mobility. These three components intertwine and show a strong research agenda in opportunistic networks. The survey shows that analytical mobility properties and abstracted graphic features are significant to the success of some routing protocols. By emphasizing on the three components, the survey will help in developing novel integrated mobility and message dissemination solutions for opportunistic networks.

1.1 Introduction

Advances in wireless communication devices bring increasing opportunities for information sharing through new mobile applications. Opportunistic networks support these applications by enabling communications between two mobile nodes when

---

X. Hong (✉)
Department of Computer Science, The University of Alabama, 101 Houser Hall,
301 Seventh Avenue, Tuscaloosa, AL 35487, USA
e-mail: hxy@cs.ua.edu

B. Gu
12800 Turtle Rock Road, Apt 12206, Austin, TX78759, USA
e-mail: bgu2687@gmail.com

B. Gu
SMC Networks, Inc., 11940 Jolleyville Road STE-120N, Austin, TX78759, USA

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_1,

they move into transmission range of each other, or when a multihop path is found between them. Such contact opportunities may spread a message among many moving nodes and eventually deliver to the designated destination. Although many of the contact behaviors are sporadic, and can be treated as stochastic processes, a person's social activities can explicitly or implicitly drive his motion, and often produce noticeable patterns, which if explored, will be of great help for message disseminations in opportunistic networks.

Often, due to mobility, short radio range, and a potential large network area, mobile nodes only form intermittent connections. A routing path may not exist between the source and the destination at an instant time. To tackle the intermittent connection problem, recent researches have proposed many routing schemes using the "store-carry-forward" principle. Inevitably, mobility has a significant impact on the success of message delivery using these routing protocols. A number of researches have treated mobility as an integral factor in their routing protocol design. For example, the abstraction of the analytical properties about mobility can be used to help message delivery. Here, mobility and mobility models play three significant roles. First, a realistic mobility model is a valuable evaluation tool. Second, analysis on mobility patterns help to develop better mobility models. Lastly, mobility characteristics help the design of routing protocols.

In this chapter, we discuss recent research in the areas of mobility models, mobility characteristics, and routing strategies in opportunistic networks. Through these three components, we emphasize the unique intertwining connections among mobility models, mobility characteristics, and routing strategies. Our classifications and discussions set this chapter apart from the survey papers in each related area. The chapter also reveals research agendas in each area, and a research trend that integrates the three areas together. Such integration helps deliver significant insights into the performance of the protocols and applications designed for opportunistic networks [37, 42].

For mobility models, we use a novel classification that captures the social role of mobile users and their geographical movements. Typically, the movement of a person is driven by his social activities and is constrained to a road surface. We classify a mobility model based on whether it uses the social intention of mobile nodes and whether it uses realistic geographic locations (maps). Early classifications have used the independence among mobile nodes (namely, the entity mobility models and the group mobility models [8]) and the degree of randomness (namely, trace-based models, constrained topology based models, and statistical models [60]). These classifications are feasible when mobility models are used as common and valuable components in simulations for protocol evaluations. For our purpose, using social role as classification criterion can better serve the evaluation need for the routing protocols that are designed taking social properties.

The further investigation is on the motion characteristics. The movement patterns, demonstrating through spacial properties and temporal behaviors, are important in forming connecting links. Also, the patterns can be abstracted to graph features. We classify existing analytical results using these aspects. These analytical results are scattered in many research papers and they have been offering insightful knowledge for realistic mobility models and routing protocols. For example, the concentration locations can help communications among nodes [46], the spacial Levy Walks mobility can help design optimal search patterns as routing strategies [47]. In this chapter, we emphasize the characteristics and the utilization of these characteristics.

For routing in opportunistic networks, we are interested in how mobility helps in generating efficient message disseminations. A routing protocol delivering better performance shows higher delivery rate, shorter delivery delay, and less energy consumption. We classify routing schemes into two main categories, namely, proactive routing and reactive routing in opportunistic networks. Different from the conventional definitions of these two terms in Mobile Ad Hoc Networks (MANETs), in this chapter, the proactive routing category describes the schemes that build upon the knowledge of all the mobile members and their movements over time; while the reactive routing describes the schemes that use collected contact information at each forwarding node. The proactive schemes could use offline and global knowledge. With reactive schemes, knowledge is collected contact by contact. Such information may be different with different mobile users. Routing decisions are made using partial knowledge. The reactive routing category is further classified into three subcategories. They are contact-based routing, community-based routing, and auxiliary node-based routing. These subcategories are based on whether the routing relies on statistics of contacts, or topology structural features (community), or auxiliary nodes and movements.

The relationships of the three components are illustrated in Fig. 1.1. The mobility models are the evaluation tools for routing protocols and the sources for analysis. Some analytical results can contribute to new mobility models with increased flexibility in reproducing the desired network scenarios. On the other hand, routing protocols can take underlying mobile topological structures from some results of mobility analysis. In summary, this chapter provides a systematical overview for mobility models, mobility analysis, and routing over recent work in opportunistic networks. And more important, it addresses the intertwining connections among the three areas. It also discusses the research trends and future directions.

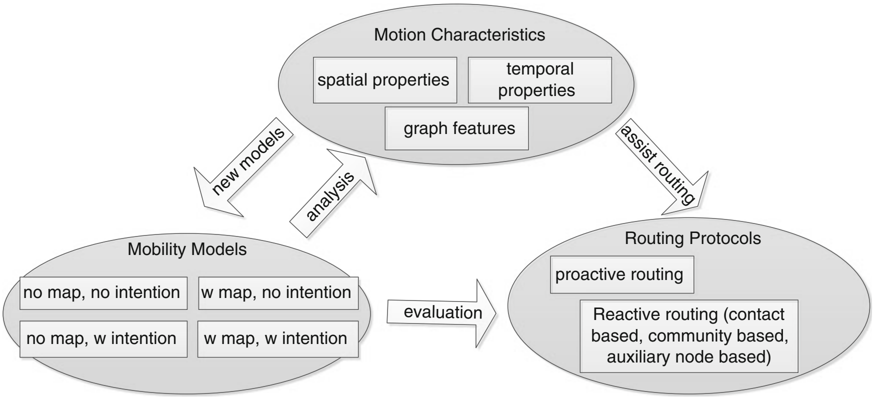


Fig. 1.1 Relationships of the three components

The rest of the chapter is organized as follows. In Sect. 1.2, we classify the mobility models into four categories with a combination of with or without intention and geographic constraints. We introduce the representative models respectively. Section 1.3 summarizes the analytical results on the motion characteristics including geographic, time related features, and graphic properties. In Sect. 1.4, brief overviews are given on several routing schemes for delay tolerant networks with a balance in coverage and focus. Future directions are discussed in Sect. 1.5. We summarize the chapter in Sect. 1.6.

1.2 Mobility Models

Users' movements are most likely the explicit or implicit results of their social or personal activities. As mobility models, the movements can map to geographical locations and motion steps. In this section, we use a criterion to classify a mobility model depending on whether it uses road systems through maps and whether it captures the intention of mobile node. A map defines the available locations and paths where mobile nodes move. For example, mobile nodes may move freely among any spots in a field or may go along on certain fixed roads. As for the intention of mobile nodes, it dictates the motion behavior and trajectories, which, in return, generate the ever-changing topology of a network. Our criterion demonstrates both the objective constraint and the subjective intention, which reflect the essential features of a realistic mobility model in a social context. In this section, the two specific factors are combined to classify the mobility models.

1.2.1 Non-Map Without-Intention Models

In the non-map-without-intention category, the mobility models have no restrictions on paths nor intention of movement for mobile nodes.

1.2.1.1 Basic Model

The first basic mobility model is the Random Walk Model [8], which is sometimes referred to as Brownian Motion. A mobile node in a field moves at a certain speed and direction for a step and then changes to another speed and direction in the next step. The step can be a specific time interval or a distance. The model is memoryless, which means the current movement of the node is independent of the previous speed and direction [22]. Random Waypoint Model [28] adds a delay factor to simulate pauses during motion. Specifically, there will be a pause time between two straight line movements. The result suggests that long pause times will lead to stable network even at high speed [6]. Random Direction Mobility Model further adds treatments dealing with the movements hitting simulation boundary [43]. These models can generate unrealistic movements, such as sudden stops and sharp turns.

1.2.1.2 Realistic Model

To model reasonable movements, the Gauss-Markov Mobility Model is proposed where the current speed and direction are obtained based on the previous speed and direction [30]. The model further improves the previous models by simulating the acceleration and deceleration of mobile nodes to some extent. Moreover, the Heterogeneous Random Walk (HRW) model is able to simulate the clustered network [40]. In the model, every node performs an independent random walk with heterogeneous speed. The significant feature of the HRW model is that the occurrence clusters are guaranteed because of different speeds and not because of "popular locations". Besides that, the model is able to produce changing connectivity over time. The model uses a closed-form expression to describe the time-stationary distribution of nodes. It can generate various clustering phenomena by tuning the corresponding parameters. The result shows that the emergence of cluster is more like at the low speed areas.

1.2.2 Map Without-Intention Models

Mobility models in map-without-intention category reveal path restrictions in nodes' movements without their social intentions. The typical models are Freeway model [1] and City Block model [14]. The vertical and horizontal tracks of freeway confine the paths of movement for mobile node. Additionally, mobile nodes take random turns at the street crossings when they travel along street grid. The Street Random Waypoint mobility model uses real maps and simulates movement details for vehicular network by considering the intra-segment mobility and inter-segment mobility on street grid [11]. The vehicular movements include car following, traffic control, and turning behavior. Another mobility model for vehicular network covers the effects of stop signs, timed traffic lights, and control on next road [44].

1.2.3 Non-Map With-Intention Models

For the category of non-map-with-intention, there is no path restriction but nodes have special movement intentions. Such intentions can be individual or can be shared with other nodes.

1.2.3.1 Group-Based Model

The Reference Point Group Mobility Model (RPGM) [22] describes groups of nodes moving according to the paths of the group leaders. The path of a group leader is defined through a list of checkpoints. Members in one group add their own randomness around its leader's path. The model is general. It allows multiple groups and different methods in defining the paths for the groups. With the RPGM model, groups can move randomly within an area; they can move structurally according to a map; they can each occupy a sub-geographical area; they can also overlap partially or totally in geographical areas. The flexility of defining various motion patterns is achieved through the ways that the checkpoints are given. On the other hand, the nodes in a group have the intention to stay close in order to accomplish special missions such as battlefield situation, disaster recovery, and convention scenarios. Thus, the moving paths of these nodes are determined by the movement of their group leader.

In [34], an interaction-based mobility model with sheep and mavericks patterns characterizes the formation and disaggregation of hot spots at random times and locations. Specifically, the sheep movement pattern models the group behavior where the population at different locations serves as the influence probability. While the nodes with mavericks pattern play the role of disaggregation, the two patterns interact and cause the evolution of hot spots.

1.2.3.2 Community-Based Model

The community-based mobility model captures the feature that a number of hosts are grouped together at a certain time [35]. The model is implemented by making use of social network theory. The relationship between people is denoted as a connectivity matrix serving as the input of the model. Additionally, different types of relationships during certain times are included so as to represent the dynamics of the community.

With the assumption that each node moves independently and a node visits its own community more than others, the community model with cyclic pattern extends the community-based model by defining the repeating time period to model revisits to the same preferable locations [54]. Normal movement periods and concentration movement periods compose the time structure in this model. Specifically, in each period there are two different epochs: in local epoch the node only moves within its community and in roaming epoch the node can move anywhere. In order to match the synthetic data generated from model to the trace data, the multi-tier community model is developed to simulate the case that the neighboring areas around the community are visited frequently as well.

1.2.4 Map-With-Intention Models

The mobility models in this map-with-intention category show realistic features such as moving along paths and with intentions.

1.2.4.1 Trace-Based Model

By analyzing real mobility traces, we can obtain valuable insights into the realistic behaviors of mobile users. Many works are inspired to use trace data gathered in real scenarios for evaluation. In [2], data from a vehicular testbed of 40 buses were used to evaluate the proposed routing protocol. The buses form a Delay Tolerant Network (DTN). The impact of human mobility on message forwarding is discussed in [9]. The analysis on the data collected from six experimental traces suggests that the distribution of the inter-contact time is heavy-tail on a large range of values. Such results contradict with the exponential decay assumption implied by most mobility models.

In return, some trace-based models make use of the characteristics extracted from the trace to reproduce the trajectories for mobile nodes. Tuduce et al. [51] present a framework describing the spatial process and temporal process for the mobility model. Their WLAN mobility model is derived from the combination of spatial distributions, session length distributions, and movement decisions. They also studied the relationship between the above parameters and the impact on mobility model. The result shows that their WLAN model can simulate the real movement with a very small error. Additionally, Weighted waypoint model is presented based on the trace analysis in the campus [23]. The model captures the preferences of visiting location, pause duration, and weights for choosing next destination. The critical parameters extracted from the trace include the pause time distribution, time-varying transition probability, and wireless network usage.

The Agenda-Driven Mobility model [59] takes initiatives of people's social and personal activities in generating mobility. Specifically, it focuses on the personal agenda of activities (when, where, and what) and combines these activities to geographic locations and movements. The main components of the model involve a series of activities during a certain period of time, a map with various social sites such as schools and malls, and motion generation algorithms. It can be used to characterize many opportunistic network scenarios. In this work, the authors used the data from National Household Travel Survey (NHTS) to extract personal travel demographic characteristics such as activities, occupation types, and driving distances.

1.2.4.2 Graph-Based Model

The Area Graph-based mobility model [4] is derived from a directed and weighted graph. The vertices denote the locations with the corresponding waiting time and the edges are the paths connecting the locations with a chosen probability. During the waiting time, the node will stay inside the vertex and move according to the random waypoint model. Once the waiting is finished, the node has to pick a path to proceed. The different probability for each path will lead to the heterogeneous distribution of nodes. Such results greatly resemble the clustering phenomenon in the realistic mobile networks.

1.2.4.3 Levy Walk-Based Model

The Levy Walk is a type of random walk in which the movement increments are distributed according to a heavy-tail distribution [48]. By analyzing trace data, the authors in [41] found that human mobility is similar to Levy walk in terms of the heavy-tail distribution of flight lengths, pause times, and mean squared displacement. The deviation may arise from the map constraints such as the buildings, roads, and traffic. Based on their findings, they present a Truncated Levy Walk model that emulates human movement patterns in outdoor mobile environments with geographical constraints. This model can produce power-law inter-contact time distribution which matches the statistic feature in trace data. The movement caused by human interests or popularity of locations people visit is also implemented in the model.

1.2.5 Summary of Mobility Models

In summary, the trend of mobility modeling has moved towards being more realistic by taking considerations of both social intentions and geographical features. Early mobility models introduce randomness to motion speeds and directions. Later works have added map constraints and/or intention constraints respectively. The trace-based models can produce both features through replaying the real data. More recently, social behaviors or biological species movements are all studied in the literature. These works further spark the need for analyzing the motion patterns—the aggregated characteristics revealed in the realistic mobility. We review these properties in the following section.

Noticeably, some models are able to generate the concentrations (or called clusters) of mobile nodes. These models can be effective evaluation tools and also play an important role for message forwarding in opportunistic networks. We recap these models here: *heterogeneous random walk model*, *time-variant community mobility model*, *weighted waypoint model*, and *area graph-based mobility model*. The implementations of the models are different. In the *heterogeneous random walk model*, different speeds of mobile nodes lead to emergence of high density areas. In the other models, the preferences of locations are used to represent the popular locations. Furthermore, when the *agenda-driven mobility model* takes the daily activities and their locations, geographic concentrations are observed in their simulation results [59].

Moreover, a more explicit approach in modeling the relationships among mobile nodes can be very useful. Some models presented here are *reference point group mobility model (RPGM)*, *interaction-based mobility model* , and *community-based mobility model*. The models consider the attractions among mobile nodes. Again, the attractions are another important reason for concentrations. In some scenarios, the concentrations can have great influence on the network topology.

1.3 Mobility Characteristics

Various mobility models have been analyzed over the years and the analytical results have contributed to performance evaluation, simulation calibration, and routings protocol design. In this section, we introduce these analyses in four categories, namely, spatial characteristics, temporal characteristics, spatial-temporal analysis, and graph characteristics. The characteristics specifically include moving distance, spatial locality distribution, temporal properties, movement correlations, and graph-related features. At the end of this section, the impact of mobility models on evaluation results is discussed.

1.3.1 Characteristics of Flight

The flight length is defined as the longest straight line trip from one location to another. The probability density distribution of flight in Levy Walk mobility model [47] is expressed as follows:

$$p(l) = |l|^{-(1+\alpha)} \tag{1.1}$$

where α has a value between 0 and 2, l denotes the flight length. When $\alpha \geq 2$, the model becomes Brownian Motion. As for the property of flight in Random Waypoint Model [28] with a rectangle area $a \times b$, the probability density function is

$$f(l) = \frac{4l}{a^2b^2} f_0(l) \tag{1.2}$$

where the detailed expression of $f_0(l)$ is shown in [3]. The flight characteristics can reflect the diffusivity of mobility which is defined as the variance of translation between two waypoints. The conclusion about diffusivity suggests that the Random Waypoint model is the most diffusive model, the Brownian Motion is least, and the diffusivity of Levy Walk model is in-between [41].

The feature of flight is also mentioned in other mobility models. Specifically, the flight during each mobility epoch in community-based model is obtained as an exponential distribution with an average length. In the social network theory-based community model [35], the communities are mapped to locations in certain topographical space, hence the flight distribution heavily depends on the underlying map and relationships among communities.

For routing purpose, flight can be applied to discover the nodes that are helpful for relaying message. For example, in order to simulate the optimal search pattern, the nodes moving similar to Levy walk are selected for relaying message. In addition, the diffusive nodes with average longer flight can be employed to implement fast message dissemination [47].

1.3.2 Locality Distribution

Different movement patterns can lead to various spatial locality distributions. In general, we say that the scattering of nodes is either uniform or heterogeneous. The uniform distribution denotes the scenario that every node has the equal chance to visit each location in the network. With heterogeneous scattering, clustering phenomenon appears at some locations.

The distributions for Brownian Motion and Random Waypoint Model are analyzed in [5]. The result shows that Brownian Motion can always produce uniform position distribution. However, the space border can make an effect on the distribution in Random Waypoint model because of its diffusivity. If the pause time or pause probability increases, the border effect becomes weak and the mobile nodes can distribute in the space uniformly. The distribution of node position in Random Waypoint model can be found in [3]. Since there are only three states in the model: static, moving, and pausing, the probability density function is

$$\begin{aligned} f(x,y) = {} & f_{\text{initial}}(x,y)p_s \\ & + f_{\text{pause}}(x,y)p_p(1-p_s) \\ & + f_{\text{move}}(x,y)(1-p_p)(1-p_s) \end{aligned}$$

where p_s is the probability of being static and p_p is the probability of pausing. The concrete derived formula is given in [3].

In the Heterogeneous Random Walk [40], the node's position in a stationary regime shows in the following pdf:

$$f(x,y) = \begin{cases} c_1 = \frac{a}{\sigma_1^2} \text{ if } (x,y) \in C \\ c_2 = \frac{a}{\sigma_2^2} \text{ if } (x,y) \in \overline{C} \end{cases}, a = \frac{1}{\frac{|C|}{\sigma_1^2} + \frac{|\overline{C}|}{\sigma_2^2}} \tag{1.3}$$

where $|C|$ and $|\overline{C}|$ are the sizes of the area C and $\overline{C}$. C and $\overline{C}$ have different nodes' density, σ_1 and σ_2. They are instantaneous variances of diffusion process which reflects the speed for each area. Since the average speed is slower in C, movement beginning with the distribution will lead to the appearance of clustering in C.

The social network-based model studies the relationship between individuals for the distribution of node positions [36]. An Interaction Matrix is used to express the social relations. The derived Connectivity Matrix can reveal clusters by identifying the groups of nonzero entries. In another work [51], analysis on the trace data shows that wireless users stay at a few locations for most of the time and only a small part of connections last for a long time. A user location at the next time step could be one of the three possible areas which are the current cell, neighboring cell, and non-neighboring cell. Similar concentration points are also identified when a taxi trace is analyzed [46]. Further, the location-dependent pause duration and preference in choosing next destination can be found in a survey-based trace data [23].

1.3.3 Temporal Characteristics

Several temporal characteristics are proposed to show the relationship between time and mobility. The probability density distribution of pause time in Levy Walked mobility model is shown in the following equation [41]:

$$\varphi(t) = t^{-(1+\beta)}, 0 \leq \beta \leq 2 \tag{1.4}$$

where t is the pause time. In Random Waypoint mobility model, the pause time of mobile node is uniformly picked from a range of time [28]. A general conclusion is that the longer the pause time, the smaller the mobility [38].

The hitting time is used to express the average time when a node starts from the stationary distribution to move into another arbitrary location. When there are several areas with higher visiting rates in the network, the hitting time for these areas is critical in selecting one for communications purpose. Additionally, the meeting time is defined as the expected time before two nodes meet with each other. The derivation of hitting time and meeting time is given with the Time-variant Community mobility model [54]. Based on the two-state Markov model and geometric distribution, the expected hitting time and meeting time are derived. The hitting time in Random Waypoint model defines the time that a host takes to move between two consecutive waypoints. The probability density function is [3]

$$f_T(t) = \begin{cases} \int_{v_{\min}}^{v_{\max}} v f_L(vt) f_V(v) dv & \text{if } t \in [0, l_{\max}/l_{\min}] \\ 0 & \text{otherwise} \end{cases} \tag{1.5}$$

where $f_L(l)$ is the probability density function of transition length and $f_V(v)$ denotes the probability density function of speed.

The inter-contact time denotes the time gap separating two contacts between the same pair of mobile nodes. Based on the observations on eight distinct experimental data sets, Chaintreau et al. show that the inter-contact time can be approximated by a heavy-tail distribution instead of a light-tailed distribution, though the latter is common to most of the mobility models [9]. The impact on packet delivery from various values of the distribution parameter is studied as well. The concept of $k - vicinity$ extends the contact between two mobile nodes to contacts that are within k-hop range [39]. The observation is that high percentage of contacts occur within k hops in mobile opportunistic networks. Building on $k - vicinity$, $k - contact$, and $k - intercontact$ are defined. The former characterizes the property that two nodes are within k hops in the $k - vicinity$, and the latter describes the inter-contact time of two nodes based on the contact states of the two disjoint k-vicinities each belongs to. These two metrics capture more contact opportunities for message sharing and also explain the social behavior of the community-based mobility model.

The authors in [45] provide an analytical framework to derive the statistics for link lifetime, new link inter-arrival time, link breakage inter-arrival time, and link change inter-arrival time. They are functions of node's speed, transmission range, and density. For the link inter-arrival time, the density can be denoted as

$$f_{\text{change}}(t) = \lambda e^{-\lambda t},$$

where λ is the mean link change rate which is a function of node velocity. This result is applied to adaptive periodic updating interval for a proactive routing protocol in mobile ad hoc networks.

Another routing relevant temporal feature is *encounter frequency*. It can be obtained from encounter history during a certain period. This statistical data is used as a metric to predict the encounter probability in the future. A couple of protocols such as [31, 49] make use of counter frequency as the simple and effective routing metrics. The above contact properties are also analyzed in [61] with special efforts for two distinct groups of people. One group of people visit a specific area periodically, while the other group shows random visits and occurs at the specific areas rarely.

With the interaction-based mobility model [34], the *filling time* and *scattering time* are proposed to describe the dynamics of hot spots. The filling time denotes the time when an expanding hot spot becomes stable and the scattering time indicates the time when the hot spot disappears. The two metrics describe the graph evolution when formation and disaggregation of hot spots change at random.

1.3.4 Joint Spatial and Temporal Analysis

The temporal and spatial properties are often studied jointly. In [29], the relationships among vehicles are analyzed by first creating a mobility profile of each node, which includes a series of times and locations of its movement. Then an entropy-based metric using the node profiles is proposed to compute the group similarity . Given a grouping threshold, the metric helps to determine whether two mobile nodes are neighbors or not at a given time.

A different approach is introduced in Mobyspace [27]. This work computes the similarity of mobile traces using a set of metrics including Euclidean distance, Canberra distance, Cosine angle separation, and Matching distance. The results associated with these metrics map each individual mobility into a high-dimension space, where the group similarity is found.

The time-dependent link properties at different locations are studied in [19]. The analysis identifies popular locations in mobile networks and trajectory segments between each pair of these locations. The corresponding time-dependent graph is formed and the aggregated mobile segments are analyzed for each link to derive the link delay and capacity. The probabilistic analysis under the similar network scenario is given in [20, 21]. In these works, the movement duration is discretized into steps and the forwarding strategy is modeled as a probabilistic value. The analysis uses matrix to calculate communication latency for different forwarding strategies.

1.3.5 Graph Characteristics

It is interesting to note that some trace-based mobility models have given rise to a new type of analysis, i.e, the potential indication of social networks. Typically, when nodes visit common places, or they move close to each other, the wireless devices can pick up the connections in proximity. In recent works, researchers have analyzed these connections to identify graph-related characteristics. The existence of nodes bearing the graph structural properties cannot be ignored since they can make a strong impact on message delivery, for example, a node showing centrality in the network can be applied to bridge the gap between two distant nodes.

Centrality [17] is used to measure the importance of node in terms of the network structure. The three common metrics are degree centrality, closeness centrality, and betweenness centrality. Specifically, degree centrality is defined based on the number of direct links of a given node. The closeness centrality measures the distance of each direct link. A shorter average distance of a node implies a higher closeness centrality. To measure the betweenness centrality, the shortest paths between any other nodes yet containing a given node are counted. Nodes with high centrality act as the central nodes. They can play an essential role of message relay. Generally, these central features are not proposed for network with dynamic topology because the derivation of central nodes needs global network information. Thus, it is necessary to search and compute the centrality locally.

Additionally, the concept of clique community is proposed to classify mobile nodes in a network for efficient routing. Specifically, the mobile nodes in the same community become the favorable message relays. A distributed algorithm [25] is provided for identifying k-clique community. The first step is to build a familiar set for each node which consists of its contact neighbors, and then the local community is generated by combining the related familiar sets subject to specific admission condition.

Given the importance of the connectivity of a network graph formed by mobile nodes, a measurement is proposed in a continuum framework [10]. The continuum space consists of two dimensions: the average node density and the average node speed. According to different combinations of density and speed, three classes of networks are found, which are the total connected network, disrupted network, and partitioned network. The building of their framework starts from each node pair and the node pairs are classified into different types based on the connection duration. Finally, the percentage of node pairs with a certain type is used to determine the whole network connectivity.

1.3.6 Simulation-Related Issues

Contact traces usually contain link information of encountering nodes but not the location coordinates of these nodes. The usage of a contact trace is thus limited to the environment where the trace is acquired. In order to improve the usability of the contact trace, e.g., transferring a Bluetooth trace for a WiFi contact environment, Whitbeck et al. [55] try to derive mobility information (position data) from a wireless contact trace. As a result, a single trace can generate more network scenarios for simulation. The authors use the constraints of speed and contact event, and an online force-based dynamic graph layout algorithm to calculate (infer) future node contacts. Their evaluation results show that reliable high frequency contact trace is helpful to derive accurate prediction. And using additional reference nodes can further increase the precision.

Several unexpected problems have emerged regarding to a certain mobility models. The issues are that the simulations using these models may bring inaccurate results in terms of evaluation. Taking the Random Waypoint model as an example, the first problem is the decay of speed [56], i.e, the average speed will decay to 0 when the speed range sets to [0, x]. Another problem is that the density at the center region will be higher than that around the boundary [3]. Jean-Yves Le Boudec applied Palm calculus to explain above problems [7]. The stationary distribution of nodes is also derived which can be used to start the simulation directly, eliminating the need for a long warm-up running time to reach the stable state.

1.3.7 Summary of Characteristic Analysis

The analysis on motion patterns has revealed many salient properties in spatial, temporal domains, and nodal graphic characteristics. These analysis results are being used in many areas. Some of the spatial and temporal properties from trace analysis are used in new mobility models, e.g., the power law distribution in flight length and flight time as in the Truncated Levy Walk model. Some results are used in analytical evaluations of latency and delivery ratio for routing protocols. The trustworthiness of simulation is also analyzed using the nodal spatial distribution for random waypoint mobility model. Warnings are given based on the results regarding the minimum speed and the treatments on the boundaries. The graph-related properties can suggest specific roles that the nodes may perform in forwarding messages. Those nodes with strong graph structural properties play an important role in expediting the forwarding and offering efficiency. Many characteristics are used by routing protocols directly, for example, interarrival time can be used to tune message broadcast interval, centrality and similarity can be used for forwarder selection, diffusivity can be used for forwarder selection as well. These analysis results are summarized in Table 1.1.

1.4 Routing Strategies

Disseminating messages with the “store-carry-forward” routing principle has been greatly studied as Delay Tolerant Network (DTN). We classify the various routing strategies into two categories: proactive routing and reactive routing. The proactive

Table 1.1 Summary of mobility characteristics

Categories	Mobility characteristics	Features for routing
Flight length	Longest straight line trip from one location to next location; node diffusivity	Message forwarder adopts Levy walk, high diffusive nodes for fast dissemination
Locality distribution	Distribution of node positions during moving process is either uniform or heterogeneous	Cluster-based routing is suitable in network with heterogeneous distribution
Temporal characteristics	Encounter frequency, pause time, hitting time, meeting time, inter-contact time, filling time, scattering time	Encounter history for choosing next forwarder; the other time metrics for estimating message delay and delivery rate
Joint spatial-temporal	Time and location relationships of groups, trajectory similarity	Routing uses clusters or high similarity nodes
Graph characteristics	Degree centrality, closeness centrality, betweenness centrality, k-clique community	Node with higher centrality as forwarder; community helps to group mobile nodes; connectivity analysis and evolution for performance

routings use the centralized or offline knowledge about the mobile network to make the routing decision. In the reactive routing, nodes derive the forwarding strategies through the contact history, without a global or predetermined knowledge of the occurrences of future (maybe better) links. Most DTN routing protocols fall into the reactive category. We further classify them based on the types of mobility characteristics or the usage of assistant nodes. It is worth noting that graph-related characteristics are regarded as important enablers for performance improvements, and thus are used by the protocols in both categories. Given the rich literature resources, we could only select representative routing protocols in our discussions.

1.4.1 Proactive Routing

Using the knowledge about the mobile network such as nodes' contacts history, queueing length and traffic demands, a network graph can be obtained in advance. The graph includes edges with time-varying capacity and propagation delay. Jain et al. [26] proposed a framework of routing which takes different levels of knowledge modeled after the graph to calculate minimum delay path for message delivery and trade-offs. The work shows that the more knowledge acquired by the routing protocol the better the performance. The authors in [2] treat the message routing as a resource allocation problem. Packets in the buffers will be forwarded based on the decisions on whether to be replicated or not in order to optimize a specific routing metric. A per-packet utility function is derived from the routing metric designed by administrator. After the information for utility is received from control channel of node, the inference algorithm in the protocol will estimate the utility for each packet and the result will be used to pick up the corresponding packet to be replicated and sent. The distributed estimation procedure can be found in [2] which assumes that the inter-meeting time between nodes is exponentially distributed.

Taken the fact that cyclic movement patterns occur in real mobility scenarios, the mobile network can be modeled as a stochastic contact graph where every node is a vertex and each link has the contact probability and expected latency [32]. In routing, Markov decision process is applied to search the path with expected minimum cost for message routing.

The cyclic movement pattern is also considered in the network scenario where static throw-boxes are deployed [19]. The mobile nodes traveling between throw-boxes form network links that carry the temporally stored messages from one box to another. Multiple cyclic movement patterns are multiplexed to form a contact graph with time-dependent metric states at each link. A capacity-aware routing protocol is proposed which searches the shortest path that considers the time-varying delay and capacity of the virtual links for each box pairs. A Markov model is used to describe the evolution of the real-time link delay and capacity between two throw-boxes. The routing decision is made for each Markov state at each link.

Mobyspace [27] describes a generic routing scheme using high-dimensional Euclidean space with the assumption of full knowledge at each node about other nodes' mobility patterns. The main routing idea is that the packet should be forwarded to the node which has the similar pattern of destination. Several metrics are proposed to compute the similarity of mobility models including Euclidean distance, canberra distance, cosine angle separation, and matching distance. Based on the results of similarity, each node can efficiently route the message to the right forwarder.

1.4.2 Reactive Routing

The reactive routing category further divides into three subcategories as the contact-based routing, community-based routing and the auxiliary node-based routing. In the contact-based routing, communication is achieved by a series of encounters among mobile nodes. Specifically, when two nodes contact with each other, they can decide to exchange messages according to a predefined metric. The community-based schemes typically identify and use a special group of nodes. Messages will be sent to this group of nodes. This special group is formed by the nodes with better sociability, frequent contacts with the destinations, or attached to a hot location. By identifying the communities, messages can be quickly transmitted to the destination. In auxiliary node-based routing, extra nodes are introduced to act as message forwarders. These nodes may move with favorable trajectories or stay at specific locations.

1.4.2.1 Contact-Based Routing

Epidemic routing [52] is the basic contact-based routing in DTN. In this scheme, when two nodes encounter, the message called summary vector is exchanged in order to detect the missing contents in each other. Once a node finds the discrepancy, it will request the unseen message. The epidemic routing will flood a message with the expense of wasting huge resources such as bandwidth and buffer sizes. Based on the epidemic routing, PROPHET [31] employs a probabilistic metric called delivery predictability to narrow the scope of message flooding. The delivery predictability indicates the likelihood of the encountering node being able to deliver the message to the destination. Three equations are used to predict the delivery probability. The equation for probability updating is:

$$P_{(a,b)} = P_{(a,b)_{\mathrm{old}}} + (1 - P_{(a,b)_{\mathrm{old}}}) \times P_{\mathrm{init}} \tag{1.6}$$

where $P_{(a,b)} \in [0, 1]$ means the contact probability at node a for destination b and $P_{\mathrm{init}} \in [0, 1]$ is an initialization constant. The equation for aging is:

$$P_{(a,b)} = P_{(a,b)_{\mathrm{old}}} \times \gamma^{k} \tag{1.7}$$

where γ is aging constant and k is the number of time units that have elapsed. The last one is to measure the transitivity which is for the case that if node A frequently meets node B and node B frequently encounters node C, then node A is a good candidate to relay message to C(through B) even if A rarely sees C.

$$P_{(a,c)} = P_{(a,c)_{\mathrm{old}}} + (1 - P_{(a,c)_{\mathrm{old}}}) \times P_{(a,b)} \times P_{(b,c)} \times \beta \tag{1.8}$$

where β is the scaling constant to decide the impact of transitivity on the delivery predictability. Therefore the forwarding happens only when the delivery predictability of neighboring node is high.

The Spray and Wait protocol [49] significantly reduces the transmission overhead of flooding-based scheme by spraying only a fixed number of copies of message into network and then wait until the nodes that carry these messages encounter the destination node. The number of copies used by the protocol is determined through an equation based on the relationship between the number of nodes and the amount of copies. The Spray and Focus is also proposed in [49] which aims at improving Spray and Wait for mobile users with localized mobility. The difference between the above two protocols is that the message carrier in Spray and Focus will forward the copy to another suitable neighbor if they have not encountered the destination for a long time. The Seek and Focus is a hybrid protocol which includes utility-based routing and randomized routing. It builds on the Spray and Focus protocol to conquer the slow-start phase problem. The initial step of this protocol discovers the potential relay neighbor by using the utility-based approach. It then uses the randomized routing in the re-seek phase in order to avoid routing jamming for a long time at local maxima of utility.

A data item and interests on the data item can be another decisional factor in message exchanging among contacts [42]. Such consideration is important for data-centric opportunistic networks. Data can be described using a set of tags; and a user's interest on data can use the same tagging attributes. A *relevance score* measures the strength of a node's interest on a data item through the matching attributes between user's interest space and data's metadata. A data item with a higher score will have higher priority and being transferred earlier. However, trade-offs exist in terms of local interest versus global interest. The work introduces five strategies that detail various preferences at the times of data exchanges. Extensive comparisons are made to understand the performance under the strategies, the mobility patterns, and the data interest model.

1.4.2.2 Community-Based Routing

Various analysis on mobility patterns have shown that a certain group of mobile nodes can be at more helpful positions topology-wise. For example, mobile nodes with better sociability or having similar location preferences or motion trajectories. Such groups can be identified through various metrics, and are called communities. Many routing protocols well utilize the identified community for passing messages to achieve better performance.

The concept of centrality can be used to identify the nodes with high betweenness centrality which measures the importance of node on the communication paths. In addition, the similarity of nodes can be measured to find the nodes closest to destination. With these two metrics, the central nodes will first forward the messages to the possible node groups that contain the destination and then some nodes in the group will hand over the messages to destination.

Daly et al. provide a distributed method to identify central nodes and utilize them to forward messages [13]. The authors utilize the idea from ego network to measure the centrality in a distributed way. In addition, the similarity between nodes is calculated based on neighbor set so as to obtain the nodes close to destination. With the centrality and the similarity, the weighted metric is defined and used to choose next forwarder. In another work [24], the forwarding algorithm makes use of the averaged degree of node as approximate metric to find the forwarder with higher centrality.

In [18], an efficient multi-casting mechanism is proposed by using social characteristics including community and centrality. It determines the message forwarders subject to certain delivery ratio. A new centrality metric considering the contact frequency among nodes is provided as the weight of each node in [25]. A distributed algorithm is used to identify the communities in the network using the metric. The transmission among communities is achieved by exchanging message through the gateway nodes belonging to multiple communities.

Island Hopping [46] relies on the cluster-based mobility model. The authors introduce a novel network entity called Concentration Points (CP), which are locations where many nodes stay for a while. Nodes can communicate within the same CP.

A mobile node will carry messages from one CP to another. The routing algorithm first discovers the whole network graph collaboratively in order to calculate a sequence of CPs to forward the message. The Last Encounter Table is used to estimate the position of destination for the shortest path. One-hop acknowledgment is used in Island Hopping for reliability. Thus a message will have a few copies at the previous CP. These copies will be suppressed after an acknowledgment is received.

The protocol [33] for VANET makes full use of the underlying road map to achieve the efficient message dissemination among dynamic vehicles. In their design, the intersection graph is created first where each road segment becomes the vertex and the intersection linking any two segments is the edge. Based on that, the corresponding connected dominating set (CDS) is derived, which develops into the information search area. At the same time, the nodes only in that CDS form a special community and become the forwarders to disseminate messages.

1.4.2.3 Auxiliary Node-Based Routing

Some of the DTN routing protocols use a set of special nodes to help message dissemination. The mobile ones are called message ferries. The trajectories of the message ferries are controlled in order to maximize the chances of message delivery. The Levy message ferries are proposed in [47], where the ferries move using the Truncated Levy Walk mobility model with a smaller value of α (inducing higher diffusivity). An optimal routing scheme, using Levy Walk searching strategies, achieves better performance for sparsely and randomly distributed targets [53].

A well-known message ferry approach is discussed in [57]. There are two different message ferry schemes described in that paper: Node-Initialed MF scheme and Ferry-Initiated MF scheme. The first scheme takes advantage of the fixed routes of ferries to collect data from mobile nodes. When the nodes periodically move close to the route of the message ferry to send message, the deviation of their original path will degrade performance on the tasks they need to finish. So the balance between performance gain in data delivery and performance degradation in assigned tasks should be considered. Another scheme is initiated by ferry in which ferry takes proactive movement to contact nodes for communication. In this scheme, each node is equipped with a long range radio which is used for contact control and a short range radio for message exchange. In addition, the goal of trajectory control of the ferry is to minimize the message drops. A follow-up work employs multiple ferries to improve the data delivery performance.

Stationary device known as throwbox is another type of auxiliary node that can be deployed in order to facilitate message exchange [58]. Based on the degree of available information, three modes for deployments are presented: contact and traffic-based mode, contact-based mode, and oblivious mode. Intensive simulations suggest that throw-boxes are effective in improving throughput and delay when multi-path routing and regular movement are employed.

1.4.3 Summary of Routing Strategies

In this section, we introduced many representative DTN routing schemes. We also tried to identify the noticeable trend that explores the underlying network mobility. The helpful mobility features demonstrated not only in geographical phenomena but also in social network implications. Brief summaries and comparisons of the routing schemes are given in Tables 1.2 and 1.3 for proactive and reactive routing protocols respectively. We use a column *Mobility Model and Feature* to reflect the influence from mobility. It includes the movement patterns and potential mobility features that the protocols rely on. The column of *Applicable Environment* expresses the scenarios where a protocol can perform better according to the original papers. As indicated in the tables, while each protocol can be applied to general opportunistic network scenarios, applicable situations exist for many protocols in order to fully utilize their design strengthes.

1.5 Future Directions

Future investigations can go in several directions. One direction is the social network-related analysis and its connection with opportunistic networks. As pointed out by Conti et al. [12], the translation from social network relationships to geographical distance bounded contacts is not a trivial issue. In this regard, mobile social networks can provide useful insights. Contact traces, on the other hand, still warrant for in-depth investigations. The work presented in [61] claims that traces from general users and all kinds of devices are highly desired. The work itself also proposes a method to identify the more socialized population at a specific area against rare and random occurrences. To be noted are the the graph-related characteristics, they shall still be of the most importance in the analysis of social behaviors. More study can investigate graph characteristics associated with temporal properties and subgraphs.

Many of the movement characteristics are closely related to the real fields where the data are obtained. As such, some most recent work has focused on vehicle movements within a road system. The unique geographical constraints and movement constraints pose great challenges to network connectivity, partition, and message persistency. These are all interesting topics for further research.

A step further will be the novel message dissemination schemes that explore more social network properties. For example, the social-aware content sharing scheme exchanges mutual interested contents when a node moves into a new community [15]. The scheme relies on closely coupled social communities to their physical device contact patterns.

Moreover, the management of opportunistic networks can take a great leap when cognitions on network traffic, radio resources, and mobility conditions are devised [50]. As illustrated in the paper, a broad range of applications can be developed. Examples include extending coverage, capacity, and traffic aggregation when

Table 1.2 Summary of proactive DTN routing schemes

Protocol	Category	Main routing strategy	Mobility model and feature	Applicable environment
Knowledge-based routing schemes [26]	Proactive	Use modified Dijkstra with cost function from different oracles	Trace data derived topology, graph characteristic of time varying link delays	Communication opportunities are known or predictable
RAPID [2]	Proactive	Estimate utility function from routing metric and route the messages with highest utilities	Vehicular DTN traces with power-law meeting probabilities, meeting time	Nodes contact with an exponential inter-meeting time
Routing in cyclic mobile space [32]	Proactive	Derive optimal routing with stochastic graph-based on Markov decision process	Trace data with cyclic pattern, contacts	Cyclic contact pattern between two nodes
Capacity-aware routing using throw-boxes [19]	Proactive	Derive optimal routing path among throw-boxes where time-varying links are formed by mobile nodes	Trace data with cyclic pattern, derived graph with contacts and time varying links	Cyclic movement pattern between two throw-boxes
Mobyspace [27]	Proactive	Forward to node having similar mobility pattern with destination	Session durations and frequencies of location visits following power-law distribution, high dimension temporal-spatial distribution	Assume the mobility pattern of destination is known

Table 1.3 Summary of reactive DTN routing schemes

Protocol	Category	Main routing strategy	Mobility model and feature	Applicable environment
Epidemic [52]	Reactive, contact	Message flooding	Any mobility model, contacts	General mobile network
PROPHET [31]	Reactive, contact	Predicting the delivery probability of relay nodes	Community-based mobility, encounter frequency	General mobile network
Seek and focus [49]	Reactive, contact	Use randomized forwarding utility-based routing and geographical routing	Community-based mobility, encounter frequency	General mobile network
Spray and wait, spray and focus [49]	Reactive, contact	Limit the number of nodes to flood message	Community-based mobility, encounter frequency	The former prefers network with diffusive nodes; the latter prefers network with localized mobility
Social network-based routing [13], Bubble rap [24]	Reactive, community	Select the central node and similar node for routing	Trace data with some nodes showing network structural properties, graph characteristics of centrality and similarity	General mobile network
Social network-based multi-casting [18]	Reactive, community	Select the central node and community in multicast	Trace data named reality [16], graph characteristics of centrality and similarity	General mobile network
Island hopping [46]	Reactive, community	Use clusters to forward message	Random walk with exponentially distributed pause and move time, spatial locality	Rely on the presence of stable clusters
Routing in VANET [33]	Reactive, community	Rely on the nodes in the CDS of the intersection graph	Trace data of taxi in urban area, graph characteristics connected dominating set	General mobile network
Scale-free Levy message ferries [47]	Reactive, auxiliary	Select the node with high diffusivity to relay message	Flight length following truncated Levy Walk, flight length	General mobile network, message ferries in Levy walk
Message ferries [57]	Reactive, auxiliary	Use message ferries to move and pick messages	Area-based model, temporal and spatial location distribution	Ferries move in proactive manner for communication
Throwboxes [58]	Reactive, auxiliary	Use static relay devices to help forwarding	Trace-based model from bus network, spatial locality	Special static nodes are needed

concentration of users in a certain service area occurs. Much work can be done in investigating the routing strategies for the cognitive tasks.

1.6 Summary

This chapter presents a survey over mobility models, analytical results on motion characteristics, and routing strategies that largely rely on mobility. The three components embedded an intertwining research agenda ranging from early separate research on mobility models till the latest results where analytical properties about mobility are abstracted and utilized in many routing protocols. Figure 1.2 illustrates the relationship of the contents covered in this chapter. At the bottom, various mobility

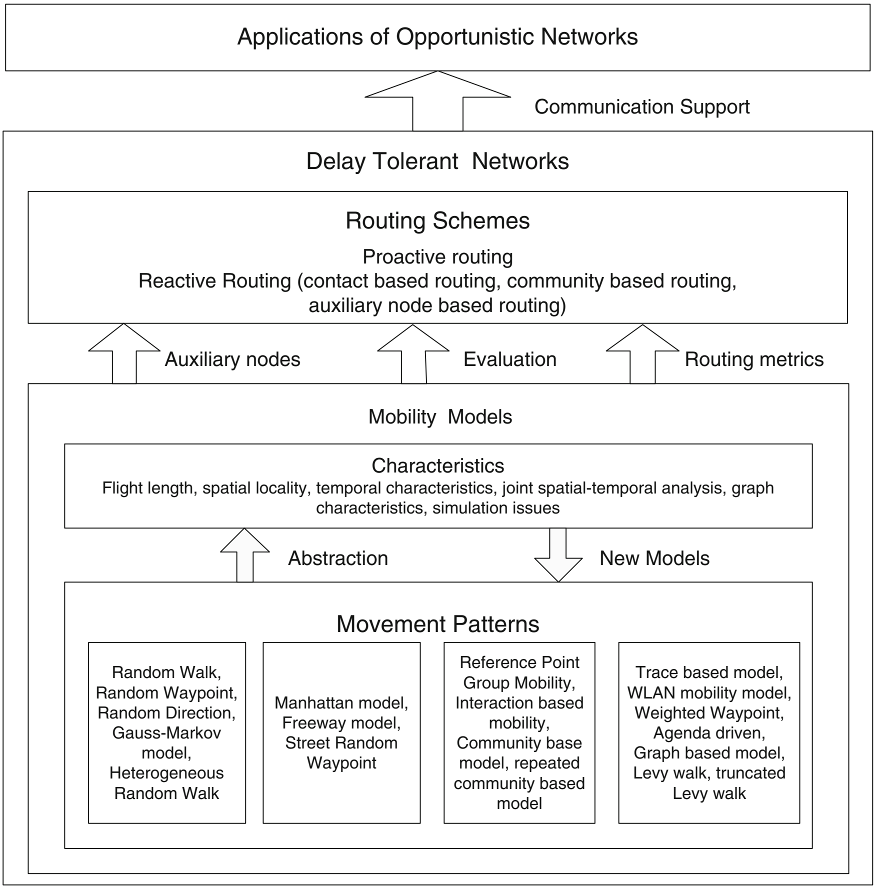


Fig. 1.2 Intertwined links between mobility and routing

patterns are listed. They are abstracted into analytical characteristics. The two parts together are under the huge scope of mobility models. The figure also shows several connections between mobility and routing schemes. On top of the routing protocols (the opportunistic network layer), distributed applications can be supported.

References

1. Bai F, Sadagopan N, Helmy A (2003) Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for ad hoc networks. In: Proceedings of the IEEE Infocom (IEEE)
2. Balasubramanian A, Levine B, Venkataramani A (2007) DTN routing as a resource allocation problem. SIGCOMM Comput Commun Rev 37(4):373–384
3. Bettstetter C, Hartenstein H, Perez-Costa X (2004) Stochastic properties of the random waypoint mobility model, Wireless Networks 10(5): 555–567 (ACM/Kluwer)
4. Bittner S, Raffel WU, Scholz M (2005) The area graph-based mobility model and its impact on data dissemination. In: IEEE PerCom
5. Blough D, Resta G, Santi P (2002) A statistical analysis of the long-run node spatial distribution in mobile ad hoc networks. In: Proceedings of the MSWiM'02
6. Boleng J (2001) Normalizing mobility characteristics and enabling adaptive protocols for ad hoc networks. In: Proceedings of the local and metropolitan area networks workshop (LANMAN), pp 9–12
7. Boudec J-YL (2007) Understanding the simulation of mobility models with Palm calculus. Perform Eval 64(2):126–147
8. Camp T, Boleng J, Davies V (2002) A Survey of mobility models for Ad Hoc network research. Wirel Commun Mob Comput 2(5):483–502 (Special issue on Mobile Ad Hoc Networking)
9. Chaintreau A, Hui P, Crowcroft J, Diot C, Gass R, Scott J (2007) Impact of human mobility on opportunistic forwarding algorithms. In: IEEE TMC, vol 6, pp 606–620
10. Chen Y, Borrel V, Ammar MH, Zegura EW (2011) A framework for characterizing the wireless and mobile network continuum. Comput Commun Rev 41(1):5–13
11. Choffnes DR, Bustamante FE (2005) An integrated mobility and traffic model for vehicular wireless networks. In: Proceedings of the 2nd ACM international workshop on Vehicular, ad hoc networks
12. Conti M, Kumar M (2010) Opportunities in opportunistic computing. IEEE Comput 43(1):42–50
13. Daly E, Haahr M (2007) Social network analysis for routing in disconnected delay-tolerant MANETs. In Proceedings of the MobiHoc 07
14. Davies V (2000) Evaluating mobility models within an Ad Hoc network. Master thesis, Colorado School of Mines
15. Delmastro F, Conti M, Passarella A (2011) Social-aware content sharing in opportunistic networks. PerAdaMagazine, pp 52–59 doi: 10.2417/2201003.002863
16. Eagle N, Pentland AS (2005) CRAWDAD data set mit/reality (v. 2005–07-01). http://crawdad.cs.dartmouth.edu/mit/reality, July 2005
17. Freeman LC (1977) A set of measures of centrality based on betweenness. Sociometry 40:35–41
18. Gao W, Li Q, Zhao B, Cao G (2009) Multicasting in delay tolerant networks: a social network perspective. In: Proceedings of the tenth ACM international symposium on mobile ad hoc networking and computing
19. Gu B, Hong X (2011) Capacity-aware routing using throw-boxes. In: IEEE global telecommunications conference, Houston, Texas, USA
20. Gu B, Hong X, Wang P (2011) Analysis for bio-inspired thrown-box assisted message dissemination in delay tolerant networks. Telecommun. Syst. (Springer, US)

21. Gu B, Hong X, Wang P, Richard B (2010) Latency analysis for thrown box based message dissemination. In: IEEE global telecommunications conference, MIAMI, Florida, USA
22. Hong X, Gerla M, Pei G, Chiang C (1999) A group mobility model for Ad Hoc wireless networks.In: Proceedings of the ACM international workshop on modeling and simulation of wireless and mobile systems (MSWiM)
23. Hsu W-J, Merchant K, wei Shu H, hsin Hsu C, Helmy A (2005) Weighted waypoint mobility model and its impact on Ad Hoc networks. In: ACM SIGMOBILE MC2R, vol. 9, pp 59–63.
24. Hui P, Crowcroft J, Yoneki E (2008) Bubble rap: social-based forwarding in delay tolerant networks. In: Proceedings of the 9th ACM international symposium on mobile ad hoc networking and computing pp 241–250
25. Hui P, Yoneki E, Chan S, Crowcroft J (2007) Distributed community detection in delay tolerant networks. In: Proceedings of the MobiArch
26. Jain S, Fall K, Patra R (2004) Routing in a delay tolerant network. In: Proceedings of the ACM Sigcomm, pp 145–158
27. Leguay J, Friedman T, Conan V (2003) Dtn routing in a mobility pattern space. In: ACM SIGCOMM Workshop on DTN
28. Johnson DB, Maltz DA (1996) Dynamic source routing in ad hoc wireless networks. Mob Comput, pp 153–181
29. Li Y, Zhao M, Wang W (2011) Intermittently connected vehicle-to-vehicle networks: detection and analysis, IEEE Global Telecommunication Conference (GLOBECOM 2011), Houston, TX, USA, 5–9 December 2011
30. Liang B, Haas Z (1999) Predictive distance-based mobility management for PCS networks. In: Proceedings of the joint conference of the IEEE computer and communications societies (INFOCOM)
31. Lindgren A, Doria A, Schelen O (2003) Probabilistic routing in intermittently connected networks. SIGMOBILE Mob Comput Commun Rev 7(3):19–20
32. Liu C, Wu J (2008) Routing in a cyclic mobiSpace. In: MOBIHOC
33. Liu K, Li M, Liu Y, Li X-Y, Li M, Ma H (2010) Exploring the hidden connectivity in urban vehicular networks. In: IEEE ICNP 2010, Kyoto, Japan, 5–8 October 2010
34. Morlot F, Elayoubi SE, Baccelli F (2010) An interaction-based mobility model for dynamic hot spot analysis. In: IEEE INFOCOM
35. Musolesi M, Mascolo C (2006) A community based mobility model for ad hoc network research. In: REALMAN ’06: Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality
36. Musolesi M, Mascolo C (2007) Designing mobility models based on social network theory. In: ACM SIGMOBILE MC2R, vol. 11, pp 59–70
37. Pagani E, Rossi GP (2012) Utility-based Forwarding: a Comparison in different mobility scenarios. In: Third international workshop on mobile opportunistic networks, Zurich, Switzerland, 15–16 March 2012
38. Perkins CE, Royer EM, Das SR, Marina MK (2001) Performance comparison of two on-demand routing protocols for ad hoc networks. IEEE Pers Commun 8:16–28
39. Phe-Neau T, de Amorim MD, Conan V (2012) Vicinity-based DTN, characterization. In: Third international workshop on mobile opportunistic networks, Zurich, Switzerland, 15–16 March 2012
40. Piorkowski M, Sarafijanovic-Djukic N, Grossglauser M (2008) On clustering phenomenon in mobile partitioned networks. In: MobilityModels’08
41. Rhee I, Shin M, Hong S, Lee K and S. Chong. On the Levy Walk Nature of Human Mobility. In IEEE INFOCOM, 2008
42. Rohner T, Bjurefors F, Gunningberg P, McNamara L, Nordstrom E (2012) Making the most of your contacts: transfer ordering in data-centric opportunistic networks. In: Third international workshop on mobile opportunistic networks, Zurich, Switzerland, 15–16 March 2012
43. Royer EM, Melliar-Smith PM, Moser LE (2001) An analysis of the optimum node density for ad hoc mobile networks. In: Proceedings of the IEEE International Conference on Communications

44. Saha AK, Johnson DB (2004) Modeling mobility for vehicular ad-hoc networks. In: Proceedings of the 1st ACM international workshop on vehicular, ad hoc networks
45. Samar P, Wicker SB (2006) Link dynamics and protocol design in a multihop mobile environment. IEEE Trans Mob Comput 5(9):1156–1172
46. Sarafijanovic-Djukic N, Piorkowski M, Grossglauser M (2006) Island hopping: efficient mobility-assisted forwarding in partitioned networks. In: SECON'06, vol. 1, pp 226–235
47. Shin M, Hong S, Rhee I (2008) DTN routing strategies using optimal search patterns. In: CHANTS'08
48. Shlesinger MF, Klafter J, Wong YM (1982) Random walks with infinite spatial and temporal moments. J Stat Phys 27(3):499–512
49. Spyropoulos T, Psounis K, Raghavendra CS (2008) Efficient routing in intermittently connected mobile networks: the multiple-copy case. IEEE/ACM Trans. on Networking 16(1):77–90
50. Stavroulaki V, Tsagkaris K, Logothetis M, Georgakopoulos A, Demestichas P, Gebert J, Filo M (2011) Opportunistic networks: an approach for exploiting cognitive radio networking technologies in the future internet. IEEE Veh. Technol. Mag. 6(3):52–59
51. Tuduce C, Gross T (2005) A mobility model based on WLAN traces and its validation. In: Proceedings of INFOCOM'05, pp 19–24
52. Vahdat A, Becker D (2000) Epidemic routing for partially connected ad hoc networks. In: Technical Report CS-200006, Duke University
53. Viswanathan GM, Afanasyev V, Buldyrev SV, Murphy EJ, Prince PA, Stanley HE (1996) Levy flights search patterns of wandering albatrosses. Nature 381:413–415
54. Wei-jen Hsu KP, Spyropoulos T, Helmy A (2007) Modeling time-variant user mobility in wireless mobile networks. In: IEEE INFOCOM
55. Whitbeck J, de Amorim MD, Conan V, Ammar M, Zegura E (2010) From encounters to plausible mobility. Pervasive Mob Comput 7(2):206–222
56. Yoon J, Liu M, Noble B (2003) Random waypoint considered harmful. In: Proceedings of Infocom
57. Zhao W, Ammar M, Zegura E (2004) A message ferrying approach for data delivery in sparse mobile ad hoc networks. In: Proceedings of the 5th ACM international symposium on mobile ad hoc networking and computing, pp. 187–198
58. Zhao W, Chen Y, Ammar M, Corner MD, Levine BN, Zegura E (2006) Capacity enhancement using throwboxes in DTNs. In: IEEE MASS
59. Zheng Q, Hong X, Liu J, Huang W, Cordes D (2010) Agenda driven mobility modeling. Int J Ad Hoc Ubiquitous Comput 5(1):22–36
60. Zheng Q, Hong X, Ray S (2004) Recent advances in mobility modeling for mobile ad hoc network research. In: Proceedings of the 42nd annual Southeast regional conference
61. Zyba G, Voelker GM, Ioannidis S, Diot C (2011) Dissemination in opportunistic mobile ad-hoc networks: the Power of the crowd. In: IEEE Infocom, Shanghai, China, pp 1179–1187, April 2011

Chapter 2
Social-Aware Opportunistic Routing: The New Trend

Waldir Moreira and Paulo Mendes

Abstract Since users move around based on social relationships and interests, their movement patterns represent how nodes are socially connected (i.e., nodes with strong social ties, nodes that meet occasionally by sharing the same working environment). This means that social interactions reflect personal relationships (e.g., family, friends, co-workers, and passers-by) that may be translated into statistical contact opportunities within and between social groups over time. Such contact opportunities may be exploited to ensure good data dissemination and retrieval, even in the presence of intermittent connectivity. Thus, in the last years, a new routing trend based on social similarity emerged where social relationships, interests, popularity, and among other social characteristics are used to improve opportunistic routing (i.e., routing able to take advantage on intermittent contacts). In this chapter, the reader will learn about the different approaches related to opportunistic routing, focusing on social-aware approaches, and how such approaches make use of social information derived from opportunistic contacts to improve data forwarding. Additionally, a brief overview on the existing taxonomies for opportunistic routing as well as a new one, based on the new social trend, are provided along with a set of experiments in scenarios based on synthetic mobility models and human traces to show the potential of social-aware solutions.

W. Moreira (✉) · P. Mendes
SITILabs, Universidade Lusófona, Campo Grande, 376, Ed. U, 1° Piso,
1749-024Lisboa, Portugal
e-mail: waldir.junior@ulusofona.pt

P. Mendes
e-mail: paulo.mendes@ulusofona.pt

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_2,

2.1 Introduction

The increasing capability of portable devices provides users with new forms of communication. They can quickly form networks by sharing resources (i.e., processing, storage) to exchange information and even share connectivity. This is possible through opportunistic contacts among devices carried by users that can forward information on behalf of other users to reach a given destination in networks with low probability of providing end-to-end connectivity between any pair of nodes, at any moment in time.

Such opportunistic communication has to cope with link intermittency. Due to such intermittency—which results from node mobility, power-saving schemes, physical obstacles, and dark areas (i.e., overcrowded with access points operating in overlapping channels or no infrastructure at all)—no end-to-end path may exist, as said, which consequently causes frequent partitions and high queueing delay.

Routing solutions based on the knowledge of end-to-end paths perform poorly, and instead numerous new opportunistic routing protocols have been proposed taking advantage of devices capabilities to overcome intermittency. Some opportunistic routing protocols use replicas of the same message to combat the inherent uncertainty of future communication opportunities between nodes.[1] In order to carefully use the available resources and reach short delays, many protocols perform forwarding decisions using locally collected knowledge about node behavior to predict which nodes are likely to deliver content or bring it closer to the destination (cf. Fig. 2.1). For that, nodes must have enough processing power and storage to keep data until another good intermediate carrier node or the destination is found [5], following a store-carry-and-forward (SCF) paradigm.

Figure 2.1 depicts how content opportunistically reaches its destination by being transferred and temporarily stored among nodes. The content is relayed by the source node to intermediate nodes, which in turn relay the content among themselves until the destination is found.

Proposed approaches range from using node mobility to flood the network for fast delivery (e.g., *Epidemic* [40]) up to controlling such flooding to achieve the same results based on: encounter history (e.g., *PROPHET* [11, 23]), optimized delivery probability (e.g., *Spray and Wait* [36]), prioritization (e.g., *MaxProp* [4]), and encounter prediction (e.g., *EBR* [31]).

Since 2007, a trend has emerged based on different representations of social similarity: (i) labeling users according to their work affiliation (e.g., *Label* [15]); (ii) looking at the importance (i.e., popularity) of nodes (e.g., *PeopleRank* [30]); (iii) combining the notion of community and centrality (e.g., *SimBet* [8] and *Bubble Rap* [17]); (iv) considering interests that users have in common (e.g., *SocialCast* [7]); and (v) inferring different level of social interactions aiming at predicting future social interactions from the users' dynamic behavior found in their daily life routines (e.g., *dLife* [28] and *CiPRO* [32]).

---

[1] For the sake of simplicity, node and user are used interchangeably throughout this chapter.

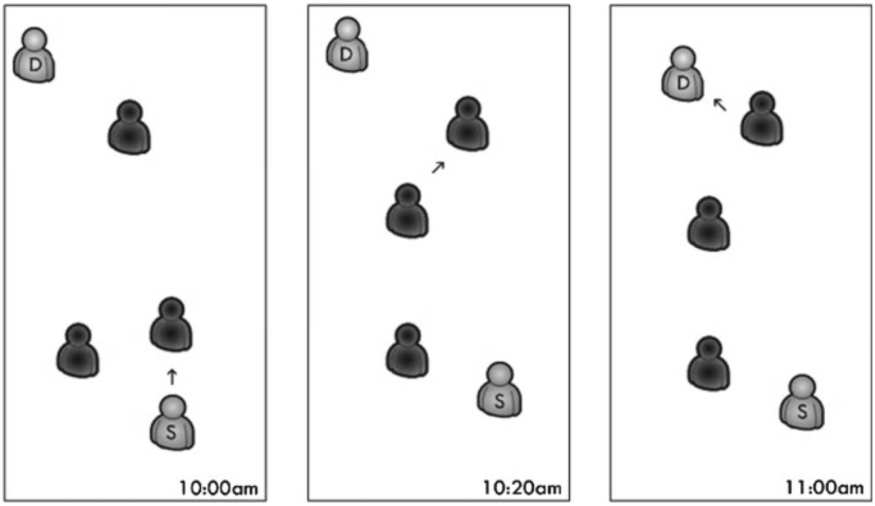


Fig. 2.1 Example of opportunistic routing

These social-aware opportunistic routing solutions have shown great potential in what concerns information delivery, since (i) cooperation among users sharing social aspects (e.g., relationship, tastes, interests, and affiliation) is encouraged, which is beneficial to improve content dissemination [18], and (ii) social information is much less volatile than human mobility, providing more robust and reliable connectivity graphs, which aids routing [14, 17].

With this chapter, the reader is expected to learn about the different opportunistic routing solutions, with emphasis on the social-aware approaches and how such approaches use social information to improve data forwarding. Additionally, this chapter provides a brief overview of the existing opportunistic routing taxonomies, allowing the reader to understand how social similarity has gained attention in the last years. A set of experiments is also presented to show that proposals based on social similarity indeed have a good potential to improve forwarding in opportunistic networks.

This chapter is structured as follows. Section 2.2 presents the relevant previous work that is oblivious to social information. Then, the social-aware opportunistic routing approaches are identified in Sect. 2.3 followed by a brief discussion on the existing taxonomies for opportunistic routing in Sect. 2.4, which makes a reference to when social similarity started to be considered for routing improvements. Section 2.5 presents some results to show the gains of the social-aware approaches over the social-oblivious ones, and Sect. 2.6 concludes the chapter.

2.2 Social-Oblivious Opportunistic Routing

According to Jain et al. [19], deterministic routing approaches are excellent from a performance point of view, since the more information a node can get from the network, the wiser its forwarding decision will be. However, the needed extra knowledge brings more complexity to the solution and even makes it impossible to be implemented due to the dynamic nature of user behavior. It was already shown that, although the optimal solution will need to have a broad knowledge about the network behavior and traffic demands [19], even the most simple oracle, called *Contact Oracle*, contains information about contacts between any two nodes, making it unrealistic as it is equivalent to knowing the time-varying characteristics of the entire network. Other oracles were defined, including a *Queuing Oracle* (i.e., information about instantaneous buffer occupancies), and a *Traffic Demand Oracle* (i.e., information about the current or future traffic demand). However, their assumptions about the network behavior is even more severe. The existence of such oracles would support deterministic routing algorithms able to compute an end-to-end path (possibly time dependent) before messages were actually transmitted. For example, if a *Contact Oracle* is available, modified Dijkstra with time-varying cost function based on waiting time would be enough to find available routes. However, the most realistic assumption is that network topology is not known ahead of time.

Based on the analysis of deterministic routing approaches, it is clear that the most suitable solution would be the one based on a local probabilistic decision, aiming to forward messages based on the opportunities raised by any contact within range. More elaborated solutions may also take other information into account to increase the efficiency of message progression toward a destination. Examples of such extra information are: history data about encounters, mobility patterns, priority of information, and social ties.

In order to understand the importance of social similarity for opportunistic routing and its applicability, this section provides an overview of the most relevant proposals to perform opportunistic routing spanning a 11-year period (2000–2011) and aims at contributing to a broad understanding of the existing opportunistic routing approaches prior to the investigation of social-aware solutions. In this study, routing proposals are grouped into three categories: single-copy, aiming to improve the usage of network resources; epidemic, aiming to increase delivery probability; and probabilistic based, aiming to find an optimal balancing between both previous categories.

2.2.1 Single-Copy Routing

In resource-constrained networks, which can occur in urban areas with high spectrum interference, opportunistic routing may lead to waste of resources when trying to deliver messages to a destination or set of destinations. Such waste of resources is mainly due to the utilization of message replication aiming to increase delivery probability.

Aiming to optimize the usage of network resources, some approaches avoid replication, forwarding messages to single next-hops based on available connectivity and some form of mobility prediction. This means that these proposals perform single-copy forwarding, i.e., only one copy of each message traverses the network toward the final destination. Such copy can be forwarded if the node carrying it decides (i.e., randomly, or based on a utility function) that another encountered node presents a higher probability to deliver the message.

Minimum Estimated Expected Delay (MEED) [20] is an example of a single-copy forwarding approach that uses contact history (i.e., connection and disconnection times of contacts) to aid forwarding. Contact history is a metric that estimates the time a message will wait until it is forwarded. A per-contact routing scheme is used to "override" regular link-state routing decision. That is, instead of waiting for nodes enclosed in the path with the shortest cost (based on *MEED* values), it simply uses any other contact opportunity (i.e., node) that arises prior to what is expected, in order to forward the message. To be able to do this, *MEED* must recompute routing tables each time a contact arrives, broadcasting this information to every other node in the network.

Spyropoulos et al. [38] present six examples where this type of forwarding is considered: (i) *direct transmission*, where messages are forwarded only to the destination; (ii) *randomized routing*, where messages are only forwarded to encountered nodes that have forwarding probability p toward the destination, where $0 < p \leqslant 1$; (iii) *utility-based routing with 1-hop diffusion*, where forwarding takes place if a utility function—based on encounter timers—of an encountered node is higher than the one of the current carrier; (iv) *utility-based routing with* transitivity, in which an utility function toward a destination is updated considering intermediate nodes that have high utility to such destination; (v) *seek and focus routing*, where *randomized routing* is used to find the best starting point toward the destination and henceforth a utility-based approach is used to find the destination; and (vi) *oracle-based optimal algorithm*, with which the future movement is known beforehand allowing optimal forwarding decisions to be taken aiming to delivery messages in a short amount of time.

From a resource consumption viewpoint, single-copy forwarding approaches are quite interesting, since they keep the usage of network (e.g., bandwidth) and node (e.g., energy, storage) resources at a low level. However, they suffer from high delay rates which, consequently, may result in a low delivery ratio. Another issue is related to the amount of knowledge that needs to be exchanged/available in order to aid forwarding, which in some scenarios generates too much overhead and may be impossible to implement.

To mitigate such problem, the following category of opportunistic routing approaches relies only on current contacts to increase delivery probability.

2.2.2 Epidemic Routing

Ubiquitous communication is a feature that has been present in our everyday life and requires messages to be delivered with high probability to their destinations, with the help of intermediary nodes able to implement the SCF paradigm.

Since networks are created by people moving around, opportunistic contacts are considered to increase delivery probability in networks with the aforementioned characteristics in a proposal called *Epidemic* [40]. With the *Epidemic* routing approach, every node in the network gets at least a copy of each message. Such a full replication strategy leads to an increase of the delivery rate. Replication of messages is done by means of summary vectors that are exchanged between nodes upon a contact. Such summary vectors contain the list of messages that each node is carrying, allowing nodes to exchange all messages that the other node is lacking. The proposal indeed increases delivery rate, since every potential forwarder has, with high probability, a copy of the message, assuming contacts with significant duration, and sufficient buffer space in each node. However, to avoid waste of resources, each host sets a maximum buffer size for epidemic message distribution. In general, nodes drop (based on a FIFO policy) older messages in favor of newer ones upon reaching their buffer's capacity. This means that the efficiency of the delivery process depends upon the configured buffer space, which may be substantially improved with the usage of Bloom filters [3]. In order to avoid messages to be replicated indefinitely, a hop count field determines the maximum number of epidemic exchanges that a particular message is subject to, being messages dropped based on the locally available buffer space. Since the number of hops toward the destination is not known in advance, setting a hop count may decrease the delivery probability. A stale-data removal mechanism would increase system efficiency by removing messages that were already delivered.

In an attempt to avoid waste of network resources, other proposals emerged based on a controlled replication approach. That is, the number of nodes which get a message copy is reduced by probabilistically choosing next nodes or by using a utility function.

2.2.3 Probabilistic-Based Routing

Probabilistic approaches are based on the estimation/prediction of what is the next best set of carriers for each message based on probability metrics, aiming to maximize delivery probability. Probabilistic forwarding protocols require node mobility patterns that exhibit long-term regularities, such that some nodes consistently meet more frequently than others over time: the mean inter-meeting time between two nodes in the past will be close to that in the future with high probability.

Within this category, proposals attempt, first of all, to optimize the delivery probability while avoiding full replication of messages. Besides this core concern, there are proposals that take into account the capabilities of nodes, the priority of messages, and the availability of resources aiming to achieve a high delivery ratio with low delay and resource consumption. Another concern of some probabilistic approaches is to use as few metadata as possible aiming to decrease concerns with respect to energy, processing, and bandwidth saving. Having considerable control overhead increases contention in the network resulting in message discarding and retransmissions.

Since 2003, different delivery probability metrics have been proposed including frequency encounters [4, 11, 23, 31, 35], aging encounters [10, 12, 37], aging messages [24, 36], and resource allocation [1, 33].

2.2.3.1 Frequency Encounters

Proposals based on these metrics have in common the fact that they rely on the knowledge about how many times nodes meet in a network. The proposals considered in this sub-category are the *Probabilistic ROuting Protocol using History of Encounters and Transitivity* (PROPHET), *MaxProp, Prediction*, and *Encounter-Based Routing* (EBR).

One of the most well-known approaches, the first being considered by the *DTN Research Group* (DTNRG) of the *Internet Research Task Force* (IRTF), is the *Probabilistic ROuting Protocol using History of Encounters and Transitivity* (PROPHET) [11, 23]. *PROPHET* uses a probabilistic metric called delivery predictability, which indicates what is the likelihood of a node to deliver a message to a destination based on its past contacts with such destination. If two nodes do not encounter each other for a while, they are less likely to be good forwarders of messages to each other, thus their delivery predictability is reduced in the process. Transitivity is a property of this predictability where if a node A regularly meets a node B and node B regularly meets node C, this implies that node C is also a good node to forward messages to node A. Delivery predictability helps to decide whether the node should replicate or not a given message (it is worth mentioning that delivery predictability was changed to cope with the "parking lot" problem as reported by Grasic et al. [11]): upon a contact, nodes exchange summary vectors that also have information regarding delivery predictability. This information is used to update their own delivery predictability vectors, which are used to make decisions about message forwarding. *PROPHET* delivers messages only to nodes which are better (in terms of delivery predictability) than the current carrier node, resulting in a reduction of the consumption of network resources and a high probability of messages delivery.

However, flooding can still occur with *PROPHET* if the message is originated in a node with low delivery predictability (i.e., node with low mobility) toward the destination, and this node only encounters nodes with higher delivery predictability.

MaxProp [4] is another probabilistic approach that uses a metric called delivery likelihood of messages, by having each node keeping track of a probability of meeting any other peer. Using an incremental averaging method, nodes that are seen frequently obtain higher delivery likelihood values over time. Each time two nodes meet, they exchange their delivery likelihood probabilities toward other nodes. Based on the delivery likelihood values computed by other nodes and by itself, a carrier of a message computes a cost for each possible path to the destination, up to n hops. The cost for a path is the sum of the probability of each contact on the path not occurring. This cost estimation, along with the hop count, is then used to order messages for scheduling and for dropping. In addition, *MaxProp* assigns a higher priority to new messages (i.e., low hop count) to increase their chance of reaching the destination faster, and tries to prevent reception of the same message twice by including a hop list in each message, and uses acknowledgments to notify all nodes about message delivery. Upon contact, two nodes exchange messages in a specific priority order: First, messages that have these nodes as final destinations; second, information for estimating delivery likelihood; third, acknowledgments to remove stale messages; fourth, messages that have not traversed far in the network; and fifth, send messages with highest priority.

By combining the estimation of message delivery likelihood with message priority and acknowledgments, *MaxProp* is able to reach good performance regarding message delivery probability and message latency with transfer opportunities limited in duration and bandwidth. However, it is considered that nodes have unlimited storage for their own messages and limited storage for the messages coming from other nodes, which has an impact on the overall performance.

It is important to mention that the delivery likelihood metric used in *MaxProp* is different from the delivery predictability metric employed in *PROPHET* in the sense that the former depends solely on the probability of nodes to meet each other while *PROPHET* depends on the probability to meet the destination itself, which means that *PROPHET* requires more state. However, *MaxProp* requires nodes to compute possible paths to the destination by concatenation of delivery probability between nodes while *PROPHET* does not require further computation, since messages are forwarded if a node has higher delivery predictability toward the destination than the carrier.

Song and Kotz proposed a prediction-based approach (hereafter referred to as *Prediction*) [35] that makes use of contact information to estimate the probability of meeting other nodes in the future. As happens with *PROPHET* and *MaxProp*, *Prediction* also uses historical contact information to estimate the probability of meeting other nodes in the future. However, unlike previous approaches, *Prediction* estimates the contact probability within a period of time, based on a metric called timely contact probability, which is used to compute the contact frequency between two nodes, as follows: the contact history between two nodes i and j is divided into a sequence of n periods of ΔT starting from the start time (t_0) of the first contact in history to the current time. If node i had any contact with node j during a given period m, which is $[t_0 + m\Delta T,\ t_0 + (m+1)\Delta T]$, the contact status of the interval I_m is set to 1. The probability of node i meeting node j in the next ΔT can be estimated as the average of the contact status in prior intervals. In *Prediction*, whenever two nodes meet, they exchange the indexes of all their messages. If the destination of a message is not the node in contact, the probability to deliver such message through that node is computed. If the probability of delivering the message via the contacted node (based on the past average number of encounters with the destination) within a defined period of time is greater than or equal to a certain threshold, the message is passed to the node in contact. This proposal presents two methods for choosing the next node: the decision can be taken by the node that is sending the message or by the one which is receiving it. In the former case, a metadata message is necessary to determine if the receiver is a good next hop. As for the latter, the receiver decides whether or not to keep the copy of the message considering its own probability of coming into contact with the destination.

Prediction achieves good performance regarding message delivery with low number of message transmission and duplication. However, its performance and storage usage is directly proportional to the *Time-To-Live* (TTL) of messages, which makes it more suitable for networks tolerant to very long delays (i.e., sparse networks).

The proposal *Encounter-Based Routing* (EBR) [31] also considers the number of times nodes meet in order to predict the rate levels of future encounters. It simply counts the number of contacts a node has with other nodes (Current Window Counter) and determines the Encounter Value (EV) that represents the node's past rate of encounters. The higher EV is, the higher the probability of successful message delivery. This also determines the number of replicas of a message that the relay node will get in each contact. Nodes maintain their past rate of encounters to predict their rate of future encounters. When nodes meet, they first update their EV values and estimate the EVs ratio, which is used to determine the number of tokens of a message replica that will be passed to each neighbor. This is a kind of time-to-live parameter also used by other approaches such as *Spray and Focus*. For security reasons, prior to EV update, nodes exchange information that will aid them to determine if their EV values are correct.

The prediction of future encounters used by *EBR* allows the improvement of latency and message delivery by reducing traffic overhead (i.e., unwanted copies). Messages are only exchanged with nodes that have high encounter rate, which avoids routes that may not result in delivery, and minimizes network resource usage. In addition, the proposal implements a security measure that avoids black hole denial-of-service attacks from malicious nodes pretending to be part of the network and announcing fake EV values. However, this approach presents some drawbacks in scenarios with multiple communities that have low rate of inter-contact times. In these scenarios, messages may be forwarded to nodes that indeed have higher EV values, but such messages will get stuck within the source community. Also, this security measure incurs in wasting contact opportunities for determining the reliability of the encountered node, and the proposal may have its performance degraded in scenarios where nodes have short contact times.

2.2.3.2 Aging Encounters

In this sub-category, the age of encounters is taken into consideration. Proposals that fall into this sub-category are: *Exponential Age SEarch* (EASE), *FResher Encounter SearcH* (FRESH), and *Spray and Focus.*

The *Exponential Age SEarch* (EASE) proposal [12], first presented in 2003, was one of the first proposals to consider history of encounters with a specific destination to support opportunistic forwarding. In addition to that, *EASE* also considers geographic position of nodes where a node would make routing decisions based on the time and location of its last encounter with every other node in the network. To be considered a good next hop, a node must either have a more recent encounter with the destination than the current holder of the message, or the node must be physically closer to the destination. Once the new hop is found, any position-based algorithm (e.g., DREAM [2]) can be employed to route the message toward it. These two phases, namely next hop search and message routing, are repeated until the best next hop is the destination itself. A second version of the proposal (*Greedy EASE* [12]) is able to change the chosen next hop during the routing phase as it checks the age of the last encounter with the destination at every hop.

EASE performs quite well in scenarios with nodes presenting random walk mobility patterns. The performance results show that the routes toward the destination have the same length (sometimes smaller) as the optimal case even for very large distances between the communicating nodes. However, this proposal is highly dependent on mobility patterns and destination speed. Its performance is easily degraded if the destination moves fast, turning the solution more costly (i.e., in terms of route length). Added to that, if mobile nodes shutdown their wireless interface (i.e., for energy-saving purposes), estimates considering location are of no use.

The *Fresher Encounter SearcH* (FRESH) [10] proposal is an example of a "blind" routing protocol, since it has no notion of coordinates. Each node keeps track of the time elapsed since the last encounter with every other node, and uses this information to choose the next hop for message forwarding. When a sender wishes to initiate data forwarding, it must first search for the next hop that is determined by the time elapsed since the last encounter between this potential next hop and the destination. This search is omnidirectional and is done in concentric rings with increasing radius until the next hop is found. It is necessary that nodes keep track of their one-hop neighbors to maintain encounter tables updated.

FRESH takes advantage of the time-distance correlation where the distance traveled during a time interval of duration t is positively correlated with t (e.g., a node met a few minutes ago is closer than a node met two hours before). With that, it is able to improve the performance of route discovery with no need for global knowledge of the network. Instead, the proposal is based on a distributed implementation where next hop search is defined in terms of local information (e.g., encounter tables). Like with *EASE*, performance depends on the mobility of nodes, as the time-distance correlation becomes noisier with heterogeneous speeds. That means that a node that has just encountered the destination may not be close to it if the destination is moving too fast. Another issue is that *FRESH* may suffer with loops in routing if source and destination are not part of a connected subset of nodes. This is, indeed, a problem especially in scenarios where isolated cluster of nodes may be formed. There is also an overhead related to the need of having one-hop neighbor encounter table updated.

The more recent *Spray and Focus* [37] approach proposes a scheme was a fixed number of copies are spread initially exactly as in *Spray and Wait* [36], which is described in Sect. 2.2.3.3. There is, however, a subtle difference: *Spray and Focus* uses only $\frac{1}{3}$ or $\frac{1}{2}$ of the L[2] messages normally used in *Spray and Wait*), but then each copy is routed independently according to single-copy utility-based scheme with transitivity [38]. In the spraying phase, ideally it would be good to be able to choose as relays the L nodes that most frequently encounter the destination. However, waiting for a "better" relay may mean that opportunities to spread extra copies are forfeited. Hence, the *Spray and Focus* scheme uses a greedy spraying phase by implementing a binary spraying algorithm (where half of the copies are shared with the encountered node) to minimize the amount of time it takes to spray all L copies, moving the problem of looking for a possibly better relay to the focus phase. In the focus phase, each potential router maintains a timer for every other node in the network, recording the time elapsed since the two nodes came within transmission range for the last time. These timers are similar to the age of last encounter [10] and contain indirect location information. For a large number of mobility models, it can be shown that a smaller timer value on average implies a smaller distance from the node in question.

Spray and Focus outperforms [37] flooding-based (i.e., *Epidemic*) and single-copy schemes (i.e., *Randomized Flooding*, and *Utility-based Flooding*) as well as the other spraying algorithm (i.e., *Spray and Wait*) under realistic mobility scenarios (e.g., modeling human behavior), by forwarding messages to nodes which have a "closer" relationship (determined by the encounter timers) with the destination. Also, *Spray and Focus* present good performance in scenarios with heterogeneous mobility using an algorithm that is able to diffuse timer information much faster than regular last encounter-based schemes. However, since its performance is highly dependent on the use of encounter timers, it can be easily degraded in scenarios where nodes are highly mobile as timers quickly become obsolete.

2.2.3.3 Aging Messages

Proposals under this sub-category have in common the fact of avoiding messages to be kept being forwarded in the network by creating metrics that define the age of message copies. *Spray and Wait*, *andOptimal Probabilistic Forwarding* (OPF) are the proposals comprised by this sub-category.

One of the first proposals wasSpray and Wait [36], which decouples the number of transmissions per message from the total number of nodes, generating a small number of transmissions in a large range of scenarios. Initially, copies of a message are spread quickly in a manner similar to epidemic routing. However, *Spray and Wait* stops when enough copies have been sprayed in order to avoid flooding, while guaranteeing that at least one copy will reach the destination with high probability. By exploiting the mobility of nodes, *Spray and Wait* operates in two steps: First, the source determines a certain number of adjacent nodes that are going to get a copy of the message (spraying phase). In the second step, the nodes that got a copy of the message deliver it directly to the destination when it gets within range (waiting

---

[2] Maximum number of copies allowed to be created per message.

phase). Each generated message can have L copies of it distributed in the network. This number of copies can be determined in two ways: (i) based on the number of nodes M and size of the network N; (ii) by estimating M when both M and N are unknown. Once L is known, the source can spread the copies of the message by passing only one copy of the message (*Source Spray and Wait*), or $L/2$ copies (*Binary Spray and Wait*) to each encountered node. In the latter case, the receiver of $L/2$ copies will spread the obtained copies in the same way. If the destination is not found in the spraying phase, the nodes holding one copy of the message at the end of the spraying phase will forward it directly to the final destination.

Spray and Wait exploits node mobility being able to limit the total number of copies and transmissions per message resulting in an energy-efficient solution with low delivery delay, although the achieved delay is inversely proportional to the number of copies. If nodes move quickly enough around the network, *Spray and Wait* shows that only a small number of copies can create enough diversity to achieve close-to-optimal delays. However, there is no acknowledgment mechanism to get rid of copies of already delivered messages and no mechanism to select the best set of forwarders in the spraying phase. In what concerns computational effort, determining L is not an easy task, since it is necessary to know M nodes beforehand and it depends on nodes performing independent random walks. This can easily result in an inaccurate measure of L, which degrades the performance of the algorithm. This problem is even worse in large dense networks with frequent disconnections and nodes following different mobility patterns.

The *Optimal Probabilistic Forwarding* (OPF) [24] protocol replicates a message upon node encounter if, by doing so, such action increases the overall delivery probability of such message. That is, if this action maximizes the joint expected delivery probability of the copies to be placed in system (i.e., in the sender and receiver nodes of the message). *OPF* aims to maximize the delivery probability based on a particular knowledge about the network, relying on the assumptions that node mobility exhibits long-term regularity (enabling the estimation of mean inter-meeting times) and that each node knows the mean inter-meeting time of all pairs of nodes in the network. *OPF* metric reflects not only the direct delivery probability of a message, such as in *PROPHET*, but also the indirect delivery probability when the node can forward the message to other intermediate nodes, as in *MaxProp*. However, unlike *MaxProp*, *OPF* metric reflects a hop-count-limited forwarding scheme, based on a function of two important states of a message: remaining hop count and residual lifetime. Such utility function may estimate the effect that message replication may have on the expected delivery rate while satisfying the constant on the number of forwardings per message (*OPF* has a performance awareness as happens with *RAPID*, for instance).

With *OPF*, every message has a residual time-to-live (T_r) that denotes a given meeting time slot, and nodes know the mean inter-meeting time ($I_{i,j}$) between any two nodes i and j in the network. This is used to determine the meeting probability ($M_{i,j}$) among any two nodes and the delivery probability (P_{i,j,K,T_r}) between nodes i and j of a message with K remaining hop count and T_r residual time to live. The delivery probability is simply given by $P_{i,d,0,T_r}$ if the message cannot use more hops to reach the destination. However, when the message is at K hops from the destination, forwarding will take place if the combined probability of the two new copies of the message at the next time-slot T_{r-1} (i.e., $1-(1-P_{i,d,K-1,T_{r-1}}) \times (1-P_{j,d,K-1,T_{r-1}})$) is greater or equals to the probability of not forwarding it at all (i.e., $P_{i,d,K,T_{r-1}}$). So when node i meets node j, whether i should forward the copy to j depends on whether replacing the copy in i with two logically new copies (i.e., in i and j) increases the overall delivery probability. *OPF* also comes in another version where K is then substituted by the number of logical tickets (L, as in *Spray and Wait*) which are going to be distributed between the two replacing copies in a message forwarding.

OPF is able to achieve good overall delivery rate with a subtle increase in delay, since it only forwards messages to really good relay nodes. Its performance is better if the relationship between nodes is greater and the hop count (K) allowed for each message is chosen wisely. However, this good performance comes with a cost, since it is really dependent on the amount of routing information available. In networks, where only local information is available due to the dynamicity of nodes, *OPF* will have its performance degraded, since it needs the mean inter-meeting time of all nodes in the network and such information may be difficult to obtain. Added to that, since the mobility model considered follows an exponential inter-meeting time, the measurements may not represent human behavior as it is known that a power law distribution better represents such behavior [14].

2.2.3.4 Resource Allocation

In what concerns, approaches that are aware of available resources, we have the *Resource Allocation Protocol for Intentional DTN* (RAPID), and *PRioritized EPidemic* (PREP).

The proposal dubbed *Resource Allocation Protocol for Intentional DTN* (RAPID) [1] opportunistically replicates messages based upon a utility function that estimates the effect that message replication may have on a predefined performance metric in a network with resource constraints. When nodes meet, they exchange metadata about messages to be delivered and acknowledgments about already delivered messages, along with messages destined to each other. With the metadata, nodes are able to determine the marginal utility (which must have the highest increase based on the performance metric considered) of replicating messages between them. *RAPID* calculates the effect of replication considering resources constraints by exchanging metadata through an in-band control channel that allows it to have a global state of the network resources (e.g., length of past transfers, expected meeting times, list of delivered messages, delivery delay estimate for buffered messages, and changed information on messages since last exchange).

RAPID has a small cost in the usage of contact opportunities due to the utilization of an in-band control channel. Moreover, the information exchanged in such channel may not always be updated due to node mobility, delivery delay, and messages not acknowledged. This cost may be very high in bandwidth-constrained scenarios with short-lived contact opportunities. Besides that, there are no overall performance guarantees since heuristics are based on sub-optimal solutions supported by one metric at a time. In addition, the performance is also related to the used mobility pattern (e.g., predictable vehicular movements) and can be degraded in scenarios with unpredictable mobility patterns.

In what concerns metric-based approaches, the *Prioritized Epidemic* (PREP) [33] proposal is also based on message prioritization and the idea of prediction. *PREP* introduces the Average Availability (AA) metric that measures the average fraction of time a link will be available in the future (i.e., the inter-node cost) and defines drop and transmission priorities (in which lower values indicate high priority) for each message. By using a regular discovery algorithm, each node finds out about its links toward neighboring nodes. According to available information about the past state of the link (i.e., up/down), a node can determine the availability of the links for future use. Then, costs are assigned to links based on their AA values and epidemically broadcasted in the network. To find the lowest cost path to a destination (or its whereabouts), the Dijkstra's shortest path algorithm is employed. Any changes to AA values will trigger link costs updates, which are again broadcasted through *Link State Advertisements*.

PREP is able to generate a gradient of message replication density that is inversely proportional to the distance toward the destination. That is, copies of messages are kept as close as possible to their destinations. *PREP* sets a drop priority where messages shall be discarded, upon a full buffer, according to a cost determined by the distance between the holder of the message and the destination. The greater the distance, the higher the drop priority. In addition, a transmission priority is also set to messages considering their expiry time and the cost previously mentioned. This allows a wiser usage of resources (e.g., storage and bandwidth) having a good effect on message delivery. However, in dynamic scenarios, *PREP* has its performance degraded, since its delivery capability is inversely proportional to the level of disruption happening in such scenarios.

2.2.3.5 Considerations About Social-Oblivious Routing

This section introduced the reader to social-oblivious opportunistic routing solutions, providing an analysis of their advantages and disadvantages. With this, we aimed at highlighting the potential of social metrics to improve routing performance, since such metrics will allow forwarding decisions to exploit social similarities, which present less volatile characteristics than mobility metrics.

It is important to note that the solutions analyzed in this section were chosen according to: (i) the number of times they have been cited (i.e., served as benchmarks); or (ii) the number of benchmarks used for their evaluation.

We must highlight that some authors may consider some of the aforementioned solutions as social aware, for considering the history of encounters, for instance. However, it is our belief that social-aware solutions are those which have much more elaborate utility functions and/or consider features that can be used to identify/classify individuals or groups of these, i.e., common affiliations, shared interests, social ties, popularity, centrality, and among others, which are further discussed in Sect. 2.3.

2.3 Social-Aware Opportunistic Routing

Within the previous set of solutions for opportunistic routing, one of the processes that may lead to significant consumption of energy, processing capability, and bandwidth, is the prediction of mobility patterns, which is quite common to all probabilistic routing approaches. One alternative may be to devise probabilistic solutions that exploit not only mobility of nodes but also their social similarities. The reason is that mobility patterns change faster (causing the appearance of unwanted traffic due to out-of-date information) than social relationships between people within a society. The fact that social relationships are less volatile than mobile behavior has been proven [14, 17] to be rather useful in forwarding decisions. Additionally, the higher social similarity, the better content dissemination is [18]. Hence, since 2007 several approaches have been investigating the exploitation of social aspects such as relationships, interests, and common affiliations, in order to improve the delivery rate while decreasing the consumption of network resources. The proposals that are analyzed in this sub-category are *Label*, *SimBet*, *Bubble Rap*, *SocialCast*, *PeopleRank*, *dLife,* and *CiPRO*.

The *Label* [15] approach was one of the first proposals to introduce social characteristics into opportunistic routing. Experiments were conducted in the IEEE INFOCOM 2006 conference where nodes were labeled telling others about their affiliation/group: this allowed nodes to forward messages directly to destinations, or to next hops belonging to the same group (i.e., same label) as the destinations. The proposal looked not only to inter-contact time distribution for all the nodes inside a group, but also to the inter-contact time distribution between two groups (i.e., friendship ties), with results showing that nodes from one group may be good forwarders for nodes in the corresponding friendship group. The *Label* proposal provided the first indication that exploiting social similarities improves delivery ratio and especially delivery cost (i.e., total number of messages and duplicates transmitted). Another observation [15] is that friendship between different communities (i.e., unusual connections among nodes of both communities) can be used to slightly improve delivery ratio.

The performance of the *Label* approach is directly related to the allowed message TTL and the mixing rate of nodes in the experimental scenario. Delivery ratio is very low in the case of messages with short TTL, and it is easily degraded if nodes do not mix well. The reason is that *Label* performs only one-hop delivery and just to nodes belonging to the same community as the destination, which means that delivery may fail if the sender never encounters members of the same community as the destination.

Some nodes may have such a behavior that makes the usage of encounters inefficient to forward messages, due to their sporadic meeting rates. For instance, a node may be involved in a highly clustered network in which none of the nodes have directly or indirectly met the destination. However, paths between clusters may be insured by nodes that form bridges based on weak acquaintance ties. In this context, the *SimBet* [8] approach proposes to forward data based on the identification of these bridges and the identification of nodes that reside within the same cluster as the destination node. The major contribution is a new forwarding metric based on ego network analysis to locally determine betweenness centrality of nodes (i.e., importance of a node in the system, defined as the number of connections between nodes belonging to different communities that cross the referred node) and social similarity (i.e., probability of future collaboration between nodes in the same community). These two social parameters are determined from an adjacency matrix that each node keeps to track contacts (direct and through neighbor nodes) with other nodes in the network. When a node i meets a node j, they exchange messages that they have to each other, and request each other list of contacts. With this list, they can update their own contact list along with their betweenness (Bet) and similarity (Sim) values. Then, they exchange summary vectors that contain the destinations to which they are carrying messages along with their updated Bet and Sim values. For each destination in the summary vector, they determine their *SimBet* utility. With this, a vector of destinations is created containing all the destinations to which the node has highest *SimBet* utility. This vector of destinations is exchanged and nodes exchange messages to destinations present in their own vector and remove such messages from their buffers.

SimBet is able to forward messages even if the destination node is unknown to the sending node or its contacts. In this case, the message is routed to a structurally more central node where the potential of finding a suitable carrier is much higher. Moreover, *SimBet* makes no assumptions about the control of node movements or knowledge of node future movements. Finally, with *SimBet* messages are forwarded solely based on locally obtained information. It works based on forwarding a single copy of each message in the network, which makes it able to reduce resource consumption, mainly buffer space and energy. It has good overall performance regarding message delivery, which is close to the performance of *Epidemic* routing, but with highly improved delivery cost (i.e., very low number of required forwards to reach destination). However, this proposal may suffer with high delay since the level of contacts (i.e., how often and with whom nodes meet) between nodes is a key aspect regarding dissemination of information. That is, if the level of contacts between nodes is low, information (e.g., Sim and Bet values, contact lists) will take longer to be updated and diffused. Another issue regarding performance is the contact time, which can have a strong influence especially in scenarios where contacts are short lived.

Another proposal known as *Bubble Rap* [16, 17] also uses node centrality along with the concept of community structure to perform forwarding. With *Bubble Rap*, nodes are grouped based on social parameters (i.e., number of contacts and contact duration) and have a local/global popularity index (obtained from betweenness centrality). Messages within the same community are forwarded using local popularity, whereas messages traversing different communities use a combination of local/global popularity to reach the final destination. In the second case, whenever the message is forwarded to a member of the destination's community, the current carrier deletes it from the buffer to prevent further dissemination. The algorithm employed in this proposal is rather simple. If a source node wishes to send a message, all it needs to do is to check if the community of the destination node is the same as its own. If so, for every encountered intermediate node, it compares their local ranks and generates a copy to the encountered node that has higher rank value. Otherwise, it passes this message copy to an encountered node belonging to the same community as the destination or having higher global rank value.

Bubble Rap considers that nodes belong to different size communities and that such nodes have different levels of popularity (i.e., rank). With this, it can mimic human relationship allowing it to achieve good overall performance regarding delivery success rate with acceptable delivery cost. The performance is even better as the number of different communities and message TTL increase showing its capability to deal with human social behavior. However, reaching a destination in a different community is quite exhaustive, especially if the source node has a low rank and its community has a high number of nodes. This will incur undesirable replication within the source community. Moreover, centrality can result in overloading (e.g., processing, energy, and buffer) popular nodes, since they are outnumbered compared to the number of global nodes. In addition, it does not always mean that a high centrality node has the best contact probability with the destination community.

Differently from previous proposals, *SocialCast* [7] shows that forwarding can be achieved not only based on the social ties and mobility patterns, but also considering the interests of destinations. The proposed routing protocol determines a utility function based on the predicted node's co-location (i.e., probability of nodes being co-located with others sharing the same interest) and change in degree of connectivity (i.e., related to mobility and representing changes in neighbor sets), which is used to calculate how good data carrier a node can be. This proposal is based on the publish-subscribe paradigm, that is, nodes publish content on the network that is received by nodes according to their subscribed interests. Nodes get copies if they have higher utility regarding a given interest than the node currently carrying messages with content matching such interest. The proposal comprises three phases: (i) *interest dissemination*, in which each node broadcasts, to its first-hop neighbors, its list of interests along with its updated utilities regarding its interests as well as the last received messages; (ii) *carrier selection*, in which a neighbor is selected as the new carrier if its utility function regarding a given interest is higher than the utility function of the current carrier node; and (iii) *message dissemination*, in which messages are replicated to interested nodes and/or passed to the new carrier.

SocialCast allows messages to reach their destinations with a very low number of replications (i.e., reduced resource consumption) and stable latency. The result is good delivery ratio with low TTL values as messages are delivered within few hops. However, the co-location assumption (i.e., nodes with same interests spend quite some time together) may not always be true [30]. Since such assumption is of great importance, the proposal is compromised in scenarios where it does not always apply.

Also considering node mobility and social interaction, *PeopleRank* [30] makes use of stable social information between nodes to decide on forwarding. As its own name suggests, *PeopleRank* sets ranks to nodes according to their social interaction, and uses this ranking to decide on the next hop for data exchange as it is known that socially well-connected nodes become the best forwarders for message delivery. This ranking process is analogous to Google's page rank system, in which the relative importance of a Web page is determined according to its links to/from a set of pages.

Thus, nodes are ranked according to their position in the social graph, i.e., considering their linkage to other important nodes in the network. In order to determine its rank, a node needs to be acquainted to its socially connected neighbors and their respective ranks. So when two nodes meet, they exchange their ranks and neighbor sets, update their own ranks, and exchange messages according to the new determined ranking. The node with the highest rank gets the messages.

The more social information is available to nodes, the better is the overall performance regarding delivery success rate. *PeopleRank* is also able to keep a low cost associated to message delivery and delay. However, it is proven that considering only socially connected nodes is not enough to guarantee a good performance level, since socially disconnected nodes are also able to forward messages and could be considered to improve performance.

Most of the previous analyzed solutions focus solely on inter-contact times [6], and still significant investigation is required to understand the nature of such statistics (e.g., power law, behavior dependent on node context) [21]. Another drawback of such approaches is the instable proximity graphs they create, which follows mobility/encounter-based social similarity metrics [17]. Instead of considering the dynamicity of mobility, the evolving feature of social network structure of nodes as they move around meeting different nodes throughout the day is what matters most to social-aware proposals. Such feature has been shown to be imperative when building proximity graphs based on social interactions [13].

Existing solutions [7, 8, 15, 17, 30] succeed in identifying similarities (e.g., affiliation, communities, and interests) among users, but their performance is affected as dynamism derived from users' daily routines is not considered.

With *dLife* [28], the dynamism of users' behavior found in their daily life routines is considered to aid routing. The goal is to keep track of the different levels of social interactions (in terms of contact duration) nodes have throughout their daily activities in order to infer how well socially connected they are in different periods of the day.

The assumption here is that the time nodes spend together can be used as a measure of the strength of the social ties among them. To achieve that, *dLife* defines two utility functions [29]: Time-Evolving Contact Duration (TECD) that measures the level of social interaction (i.e., social strength, $w(a, b)$) among pairs of nodes during their daily routine activities; and TECD Importance (TECDi) that measures the user's importance, $I(x)$, considering its node degree and the social strength toward its neighbors. Upon a contact, nodes exchange their social weights toward other nodes as well as their importance: replication only occurs if the encountered node has a stronger social relationship with the destination or higher overall importance in the system than the current carrier of the message for that specific time period.

Despite of achieving good overall performance in terms of delivery probability and cost (with a delay increase trade-off), *dLife* does not reach maximum delivery performance. It is believed that the introduction of some level of randomness while choosing next forwarders may improve its performance, as randomness has been proved to increase delivery [30].

The *Context Information Prediction for Routing in OppNets* (CiPRO) [32] solution also takes advantages of what is happening in the daily routines of nodes. With *CiPRO*, the message carrier has enough knowledge about the time and place it will meet other nodes to forward the message. Such knowledge is defined in terms of node profile (with the carrier's—name, residence address, workplace, nationality, ...—and device's—battery level, memory, ...—information) which is used to compute encounter probability between a node and destination $p_{[N,D]}$ through a profile match. As in *dLife*, *CiPRO's* encounter probability can also reflect encounters happening in for specific time periods, P_i, which is given by a ratio between the sum of the probability of all the encountered nodes E by the node toward the destination D, $p_{[E,D]}$, and the set of these nodes, $|N_i|$. When two nodes meet, *CiPRO* triggers a control message which is propagated up to its two-hop neighbors and contains the node profile information of the destination in a evidence/value pair format. Thus, it will forward messages according to the type of contact, namely occasional (where neighbors with higher encounter probability—computed based on the control message—toward the destination will get the message) and frequent (where the node will use the history of encounters to a given destination to foresee a specific time period for efficiently broadcasting control packets and the message itself). For the latter, a BackPropagation Neural Network model is used to determine the future encounter probability, P_{pred}.

Like *dLife*, *CiPRO* also has a performance tradeoff between delivery probability/cost and delay, and perhaps introducing some randomness, in the occasional contact case, could bring more improvements to the solution.

In summary, proposals that take into consideration the idea of social similarity perform quite well when compared to algorithms simply based on history of encounters, encounter prediction, and message prioritization. However, most of the social-aware proposals consider the dynamism found in node mobility, instead of focusing on the dynamism of the social information that come from such movements. In addition, Hossmann et al. [13] show that, if the contact aggregation considered for determining the proximity graph is based on time window, some social metrics (e.g., betweenness centrality, similarity) can lead to node homogeneity regarding the given metric. That is, as network lifetime increases, these nodes may have the same characteristic (i.e., popularity) which will result in great impact on the performance of forwarding algorithms. This suggests that social-aware solutions must be carefully designed not to become a mere random-based solution. Moreover, most of the aforementioned proposals only support point-to-point communication. (Multi)point-to-multipoint communication is a desirable feature in opportunistic routing as it can help reaching more nodes interested in the content of the messages with better performance and wise use of resources [42].

Next, presenting the existing taxonomies of opportunistic routing, identifying the appearance of the social-aware branch, and providing a simplified taxonomy proposed in 2011 with a minor update given the appearance of new social-aware opportunistic routing solutions.

2.4 Taxonomies

The analysis of opportunistic routing approaches shows the existence of different trends based on distinct goals. On the one hand, we have single-copy forwarding approaches aiming to optimize the utilization of network resources. On the other hand, replication-based approaches aim to optimize delivery probability. Forwarding has the advantage of using network resources properly, but may end up taking too long to delivery the messages. Replication-based approaches present a message delivery probability that is, in almost every solution, very close to optimal, but with a high cost. Aiming to achieve a good balance between a high delivery probability and a low utilization of network resources, several proposals try to avoid flooding the network, by exploiting mobility of nodes, history of encounters, and social parameters.

Upon what is available in terms of opportunistic routing solutions, there are different classifications: considering aspects (e.g., level of knowledge) that led to an unbalanced classification, assigning most solutions to a few set of categories, or to very specific classification branches (e.g., by considering for instance information coding or methods to control movement of nodes). Independently of the classification approach used, our goal is to provide the reader with an overview of the existing opportunistic routing taxonomies in order to show when social similarity started to be considered and that its importance is recognized as it still appears evident in the latest taxonomies.

2.4.1 Existing Taxonomies

The first taxonomy for opportunistic routing was proposed by Jain et al. [19] based on three types of classification. The most important one is the first type of classification, which divides opportunistic routing according to the knowledge about the network that nodes need to have to perform message forwarding. The second and third classifications follow a trend already used to classify other type of routing: the second approach classifies routing as proactive (i.e., route computation happens prior to traffic arrival) and reactive (i.e., route computation takes place upon the need for sending data); the third approach classifies routing as source based (i.e., the complete route is determined by the source), or hop based (i.e., the next hop is determined in every traversed hop).

In what concerns the knowledge-based taxonomy, knowledge about the network is provided by centralized *oracles*. Four different oracles are proposed: (i) *Con-*

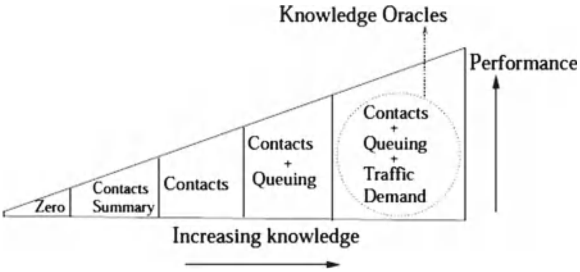


Fig. 2.2 Knowledge oracles (Jain et al. [19])

tact Summary Oracle, which provides summarized information about contacts (e.g., average waiting time until next contact, average number of contacts); (ii) *Contacts Oracle*, which provides more detailed information related to contacts between nodes at any point in time (e.g., number of contacts in a given period); (iii) *Queuing Oracle*, which provides information about buffer utilization at any time; and (iv) *Traffic Demand Oracle*, which provides information about present or future traffic demand. According to the authors, the more knowledge a routing solution can get, the better its performance will be, which leads to increasing unrealistic approaches since such ubiquitous knowledge is impossible to get in a dynamic network.

The knowledge levels used by each oracle can be: zero, where solutions use no information about the network to perform routing; partial, where solutions can route by using only the *Contact Summary* oracle, or one/both of the *Contacts* and *Queuing* oracles; and complete, where all oracles (*Contacts*, *Queuing*, and *Traffic Demand*) are considered. Figure 2.2 illustrates all the different levels of knowledge used in this taxonomy as well as the existing oracles, and the relationship between performance and knowledge.

In realistic scenarios, existing routing solutions are, however, classified as having zero or partial knowledge. This is due to the nature of opportunistic networks where network topology is not known beforehand, which makes it difficult to have a central entity (e.g., oracle) providing information to aid routing decisions. Not to mention the delay incurred to gather/process such information, which can become quite unfeasible in scenarios with short-lived contacts between nodes. Regarding the time and place for routing decisions, most of the solutions can be seen as hop-by-hop reactive approaches.

In our opinion, a taxonomy based on network knowledge, as well as proactive/reactive and source/hop characteristics is not the most suitable one to provide a balanced classification of opportunistic routing approaches.

The most complete taxonomy for opportunistic routing, to our knowledge, is provided by Zhang [41]. This taxonomy follows the work presented by Jain et al., classifying protocols according to the information they require from network as well as relevant routing strategies (e.g., pure forwarding, estimation of forwarding probability). The result is a taxonomy based on two categories, namely deterministic and stochastic routing. In the former case, node movement and future connections are known beforehand (i.e., nodes are completely aware about topology), which follows the oracle-based taxonomy proposed by Jain et al. In the case of stochastic approaches, the behavior of nodes and network is random and unknown. Hence, routing decisions depend upon local conditions leading to simple solutions where messages are replicated on every contact up to more complex solutions where the use of history of encounters, node mobility patterns, and message coding are considered for routing decisions. Zhang also classifies proposals regarding whether (or not) node movement can be controlled.

We can say that Zhang's proposal complements the proposal of Jain et al. by including a set of more realistic (stochastic) approaches. However, Zhang's proposal emphasizes aspects that are orthogonal to different routing categories (i.e., coding methods) and puts emphasis on categories that are specific to deterministic networks (e.g., node movement control), which do not represent a scenario where opportunistic routing may have greater impact in our daily life.

Balasubramanian et al. [1] propose a taxonomy based on two types of classification criteria regarding the routing strategy and the effect on performance metrics.

Based on the first criterion, solutions are divided into two routing strategies: (i) replication based strategy, where messages are replicated and then transferred to the next hop; and (ii) forwarding based strategy, where only one copy of the message traverses the network. The second classification criterion is used to divide solutions based on the effect that routing decisions will have on the performance metrics. There are two types of effects: (i) incidental ones, where the effect of decisions do not take the resource constraints into consideration; and (ii) intentional ones, which determines the effect of a routing strategy on the metrics considering such constraints.

We believe that classifying routing solutions according to their routing strategy is the most suitable approach. However, there is more to it than what is presented by Balasubramanian et al. since there are other performance metrics, besides incidental/intentional usage of resources, that can be used to distinguish among solutions that follow the same generic strategy (i.e., replication).

The same idea of classifying solutions based on their routing strategy (i.e., forwarding/replication) is also followed by Song and Kotz [35] as well as Nelson et al. [31]. In relation to the work presented by Balasubramanian et al., the classification presented by Song and Kotz is able to further divide replication-based proposals taking into consideration, not only the effect that they have on network resource consumption, but also on delivery probability. Still in what concerns the classification of replication-based approaches, Nelson et al. propose to divide them into flooding based and quota based. This classification is quite important, since its shows that it is not the fact that messages are duplicated that may lead to a flooding situation, as the one that occurs with epidemic routing, but the routing metrics used for deciding on replication. Nelson et al. show that there are probabilistic-based replication strategies that actually end up flooding the network (flooding based), since at the end of the experimental period every node has at least one replica of each message, while others (quota based) have higher success in controlling the number of replicas in the network. However, this taxonomy is incomplete in the sense that it does not consider routing categories based on metrics such as encounter number and resource usage.

In another survey, D'Souza and Jose [9] classify routing solutions into three major categories: (i) flooding based, where nodes flood the network to increase delivery probability or apply some measures to control flooding by bounding the number of messages copies and by embedding additional information to messages in order to reduce flooding effects; (ii) history based, in which the history of encounters between nodes is taken into account to improve routing decisions; and (iii) special devices based, where stationary or mobile devices are used to improve communication among the communicating nodes. Such special devices can also consider social interaction among nodes to perform routing decisions. Like Zhang, D'Souza and Jose consider aspects that are orthogonal (e.g., network and erasure coding) to any category. Despite the fact that the first social-aware solution appeared in 2007 (i.e., Label [15]), such category of opportunistic routing only made it to taxonomies in 2010 with D'Souza and Jose's work. Still, their taxonomy includes this trend under a category (i.e., special devices based) that does not comply with the regular behavior (i.e., random and unknown) found in opportunistic networks. We believe that distinguishing proposals according to whether or not they use special stationary/mobile devices to improve data exchange is not realistic as the network/nodes will have to present a deterministic behavior in order to correctly place these devices in the system.

The classification proposed by Spyropoulos et al. [39] groups opportunistic routing proposals according to the message exchange scheme they employ: forwarding (only one message copy traverse the network); replication (message is replicated in different levels ranging from every node getting a copy up to more elaborate solutions based on utility functions); and coding (where messages can be coded and processed at the source or as they travel throughout the network). The authors also identify the different types of utility functions that can be applied to either forwarding or replication message schemes. Such functions are categorized according to their dependency on the destination (i.e., destination dependent/independent). Additionally, they classify Delay–Tolerant Networks (DTN) according to characteristics that have major impact on routing such as connectivity, mobility, node resources, and application requirements. The authors succeed in mapping the routing solutions to the different types of DTNs. Still, the proposed opportunistic routing classification considers categories that can be orthogonal (i.e., coding) to any category and does not include the social similarity trend observed in 2007. They simply refer to the social aspects as a mere destination-dependent function that comprises a new research direction including social relationships, interests, and popularity to achieve suitable delivery probability with shorter delay and cost.

Figure 2.3 summarizes the analyzed taxonomies. Such figure shows an evolution toward stochastic approaches that do not require any knowledge about the global network topology. Deterministic approaches, based on some kind of centralized oracles, are not realistic and do not reflect the behavior found in opportunistic networks. Within stochastic approaches, and since 2007, there is a clear trend to classify routing strategies considering their success in achieving a good balance between delivery probability (e.g., message replication) and usage of network resources (e.g., forwarding) by employing the social similarity approach.

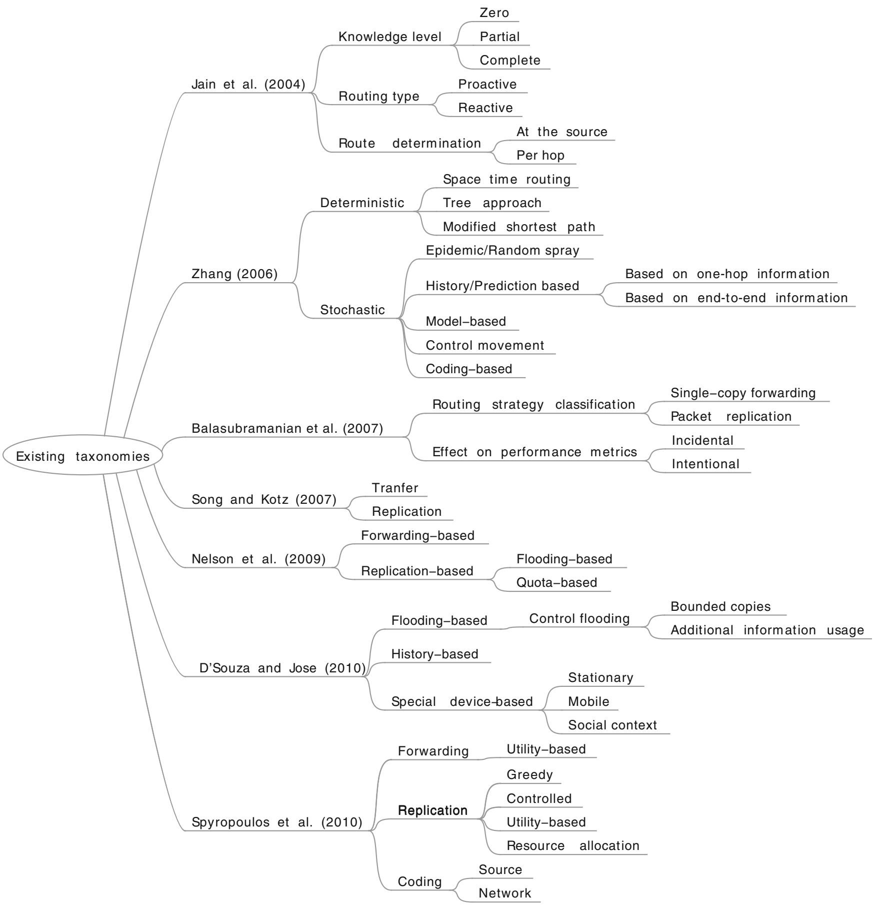


Fig. 2.3 Routing taxonomies from 2004 to 2010

Keeping in mind the ultimate goal of balancing performance and resource usage, current taxonomies are not capable of illustrating the different families of routing metrics that have been used to devise replication-based approaches able to avoid network flooding. Next, we present a simplified proposal to extend the taxonomy (presented by Nelson et al.) that better reflects the behavior of opportunistic networks with a set of categories that represent recent trends in stochastic opportunistic routing.

2.4.2 New Taxonomy Encompassing Social Awareness

One can conclude that the existing taxonomies generally focus on the analysis of opportunistic routing proposals based on their efficiency (e.g., level of knowledge employed to achieve higher delivery rates [19]; forwarding schemes that result in different performance levels [1, 35, 41]; limiting the number of messages copies in the network to spare resources [9, 31]; application of the correct routing algorithm according to the specificity of the network [39]), but they lack an analysis of the characteristics of the used graph structure.

We believe that focusing on an analysis of the topological features (e.g., contact frequency and age, resource utilization, community formation, common interests, and node popularity) assumed by each proposal may lead to a more stable taxonomy useful to study real-world networks such as computer networks operated based on social behavior.

Thus, in this section, we present a new taxonomy including updates based on the emergence of recent opportunistic routing proposals (cf. Sect. 2.3).

Social similarity is an example of recent metrics that have clearly created a new trend in the investigation of opportunistic routing [7, 8, 15, 16, 28, 30, 32]. The reason for this is that social behavior takes into account human relationship characteristics such as contacts with other people, time spent with these people, the level of relationship between people, among others. And, since computing devices are carried by humans, social-based forwarding decisions can consider people's socially meaningful relationships, where the relevant information come from aspects such as human mobility, interaction, and social structures. This information can be used to perform forwarding, because the topology created from human social behavior varies less than the one based on mobility, and thus such solutions deserve being categorized.

It is important to mention that all studied opportunistic routing proposals take advantage of node mobility to forward data ahead, where some of them (e.g., *Epidemic* [40], *Direct Transmission* [38]) are rather simple and use the resulting contacts to reach the destination, while others are more elaborate and consider social aspects in order to find the destination (e.g., *LABEL* [15], *PeopleRank* [30]). This is the reason why Moreira et al. [26, 27] do not devote a specific category to mobile-based proposals (as *Model-* or *Control Movement-based* in Zhang [41] and *Mobile Device-based* in D'Souza and Jose [9]) since this is an inherent feature of opportunistic proposals and instead look for features that help understanding their graph structure.

The taxonomy proposed by Moreira et al. [26, 27] (cf. Fig. 2.4) is based on an initial classification of all proposals as forwarding-, flooding-, or replication based. The forwarding-based category is also known as single-copy forwarding (e.g., *MEED*, and approaches in Spyropoulos et al. [38]) since all approaches propose that only one copy of each message traverses the network toward the destination. From the resource consumption viewpoint, this category of approaches is quite interesting since it keeps network (e.g., bandwidth) and node (e.g., buffer space) resources usage at a low level; however, all approaches suffer in general from high delay rates that, consequently, result in a low delivery rate.

Nelson et al. propose, depending on the level of duplication, to divide algorithms within the replication-based approach into flooding based and quota based. The flooding-based algorithms are able to increase delivery rate to a very high level, whereas the quota-based algorithms, in general, allow a more wise usage of resources, resulting in low delay and reduced flooding overhead since they tend to spread less copies of messages in the network.

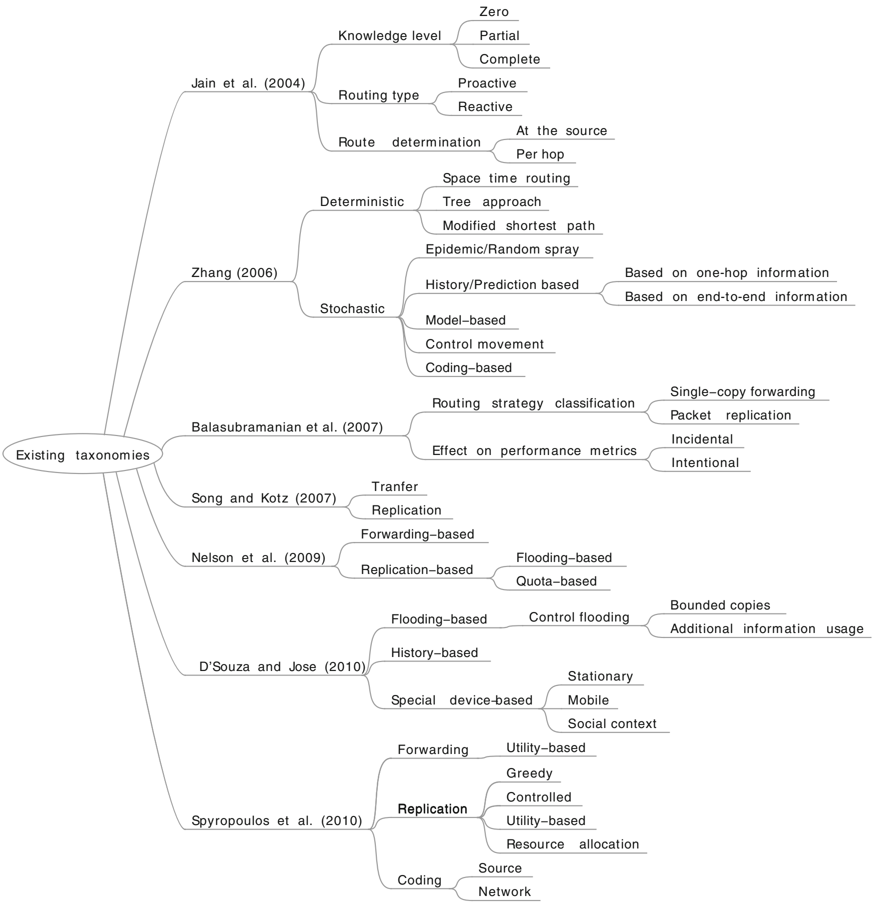


Fig. 2.4 2011 taxonomy for opportunistic routing

Moreira et al. do consider these different levels of replication, but unlike Nelson et al., they proposed flooding-based algorithms to be classified out of the replication branch. First, because only proposals that allow every node to spread a copy of each message to every other node that they meet (e.g., *Epidemic*) are considered. And, also due to the fact that having (or not) the quota-based feature (i.e., where the number of created copies does not depend on the number of network nodes) can be found in the different algorithms identified in Sect. 2.2.

Despite being an aggressive approach, the flooding-based strategy is able to increase delivery rate, but at the same time leads to a high consumption of resources. Such waste of resources can be avoided with algorithms that try to somehow control flooding. This control starts by limiting the number of copies injected in the network: if it is able to avoid nodes ending up with a replica of every created message, the algorithm has the quota-based feature.

So replication-based approaches have as common goal an attempt to increase the delivery rate by sending several copies of the initial message through different nodes to quickly reach the destination before message expiration time. Since these approaches consider different routing algorithms and metrics, they can be divided into the following sub-categories: encounter based, resource usage, and social similarity.

The first sub-category is the encounter based, where nodes choose next hops based either on frequency encounters (e.g., *PROPHET*, *MaxProp*, and *Prediction,* and *EBR*), or aging encounters (e.g., *FRESH*, *EASE*, and *Spray and Focus*). In the former case, proposals consider the history of encounters with a specific destination to support opportunistic forwarding of messages, or the frequency nodes met in the past, to predict future encounters. As for the latter, proposals consider the time elapsed since the last encounter with the destination to decide about next hops.

Resource usage is the second sub-category, in which decisions are made considering the age of messages (e.g., *Spray and Wait*, and *OPF*) or knowledge about local resources (e.g., *PREP*, and *RAPID*). Proposals based on aging messages proposals have in common the fact that they aim to avoid messages to be kept being forwarded in the network by creating metrics that define the age of message copies. As for the resource allocation proposals, they take forwarding decisions that wisely use available resources.

The last sub-category is related to social similarity, where proposals start to follow more complex algorithms aiming first at avoiding flooding with high probability, and exploiting social behavior. Thus, social similarity algorithms are divided into: community detection, shared interest, and node popularity.

Community detection approaches (e.g., *SimBet*, *Label, Bubble Rap*) rely on the creation of node communities taking into consideration people social relationships translated to contact numbers and duration of contact among nodes. These approaches suffer with the overhead of community formation. The shared interest approach (e.g., *SocialCast*) relies on the assumption that nodes with the same interest as the destination of the message are good forwarders since they have high probability to meet. But this assumption may not always be true [30], since a node with similar interest to a given group may not even come in contact with this group of nodes. Still within the social similarity category, there are approaches that are based only on a process of ranking people in terms of their popularity without a straight dependency upon neither the computation of communities nor the synchronization of interests. Node popularity approaches (e.g., *PeopleRank*) make use of social information to generate ranks to nodes based on their position on a social graph, using such ranking to decide upon the next hop for data exchange. Although social similarity algorithms provide stable graphs, it is proven that relying on socially connected nodes may not be enough to guarantee a good performance, which can be improved with the inclusion of some degree of randomness in the forwarding decision [30].

With the appearance of opportunistic routing solutions which take into account the dynamism of user social behavior (i.e., *dLife* and *CiPRO*), we propose an update to the taxonomy of Moreira et al. [26, 27]. So Fig. 2.5 illustrates the proposed taxonomy that complements the most recent trend (replication vs. forwarding) with

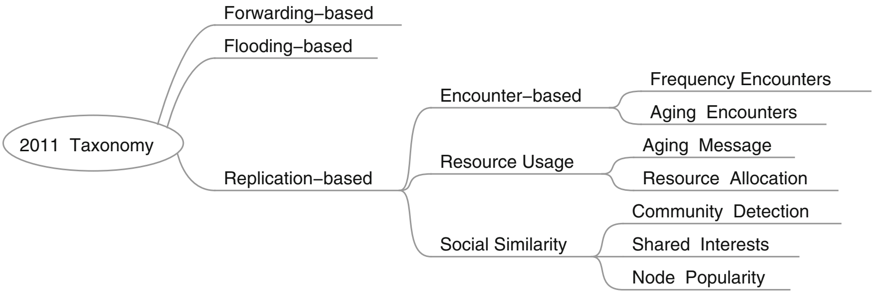


Fig. 2.5 2012 taxonomy for opportunistic routing

the analysis done of 22 proposals published between 2000 and 2012 and includes the new sub-category, user dynamic behavior.

It is easily observed that all categories of the presented taxonomy have advantages and disadvantages. However, our goal was not to identify the winner category, but to: (i) present the reader with reasons to support the emergence of the new trend based on social similarity; and (ii) update a previously proposed taxonomy with a new sub-category based on the latest social-aware opportunistic routing proposals.

2.5 Experimental Analysis

We start this section by describing the employed evaluation methodology and simulation settings. Then, we present our considerations on the obtained results pointing out the advantages and constraints of using social similarity to perform forwarding in opportunistic networks.

The results are presented in two parts: First, we show the results in a scenario with synthetic mobility models (hereafter referred to as heterogeneous scenario) and in a scenario based on real human traces; then, we present a performance comparison between some of the aforementioned opportunistic routing proposals to support our case: social similarity is indeed a trend and has great potential in improving opportunistic routing.

2.5.1 Evaluation Methodology

Experiments are done based on the Opportunistic Network Environment (ONE) simulator [22].

For that, we run simulations representing a 12-day interaction period (with 2 days of warmup, not considered for the results) in a heterogeneous scenario. For the human trace-based simulations, we considered the Cambridge traces [34]. Each simulation is run 10 times (with different random number generator seeds for the used movement models) in order to provide results with a 95 % confidence interval. All results are analyzed considering the average delivery probability (i.e., ratio between the number of delivered messages and total number of created messages), average cost (i.e., number of replicas per delivered message), and average latency (i.e., time elapsed between message creation and delivery).

Regarding the ONE simulator, the time step size is of 2 s for the heterogeneous scenario and of 1 s for the trace-based scenario for more precision in the collected data. All simulations are performed in batch mode with 2 GB RAM dedicated memory.

2.5.2 Simulation Settings

The heterogeneous scenario simulation settings aimed at creating a scenario that is as close as possible to the heterogeneous environment (in terms of mobility) that people find in their daily activities in a city, such as pedestrian walks, combined with bus and car rides, as well as people attraction to diverse locations such as work, shopping places, and home.

Our simulation scenario is part of the Helsinki city (available in ONE) and has 150 nodes distributed in 17 groups (8 groups of people and 9 groups of vehicles). All nodes are equipped with one WiFi interface (11 Mbps/100 m). One of the vehicle groups, with 10 nodes, follows the *Shortest Path Map-Based Movement* (SPMBM) mobility model, based on which nodes randomly choose a point in the map and use the shortest path to reach it. These nodes represent for instance police patrols. They move with speed between 7 and 10 m/s and have a waiting time between 100 and 300 s when arriving at the chosen destination.

The other eight vehicle groups represent buses that cover different parts of the city. Each group is composed of two vehicles each. They follow the *Bus Movement* mobility model with speeds between 7 and 10 m/s and have waiting times between 10 and 30 s.

Regarding groups of people, they follow the *Working Day Movement* (WDM) mobility model with walking speeds ranging from 0.8 to 1.4 m/s. People may also use buses to move around the city. Each group has different meeting spots, offices, and home locations. People spend 8 h at work and present 50 % probability of having an evening activity after leaving work. In the office, nodes move around and have a pause time ranging from 1 min to 4 h. Evening activities can be done alone or in a group, with a maximum of three people each. Each evening activity can last between 1 and 2 h.

For the human trace-based simulations, the Cambridge trace was considered which corresponds to a 2-month imote communication between 36 students carrying these devices throughout their daily activities.

The traffic load used in the simulations comes from a file previously generated, as has established source/destination pairs, where approximately 500 messages are generated per day among a subset of node pairs. In a simulation of 12 days (i.e., heterogeneous scenario), with 2 days of warmup time, this results in a total of 6,000 messages, from which 5,018 messages are considered for the performance assessment. As for the trace-based simulations, all the 6,000 messages are considered in the assessment.

Message TTL values are set at either 1, 2, and 4 days as well as 1 and 3 weeks. Since we want to bring our experiments as close as possible to the real world, we chose values that can represent the different applications which cope with opportunistic routing. Message size ranges from 1 to 100 kB. The buffer space is of 2 MB as users may not be willing to share all of their storage space. Message and buffer size comply with the universal evaluation framework that we proposed previously [25–27] based on the evidence that prior-art on opportunistic routing (19 proposals from 2000 to 2010) follows completely different evaluation settings, making the assessment a challenging task.

The proposals considered for our experiments comprise social-oblivious as well as social aware solutions. As representative of the former class of opportunistic routing, we chose *Epidemic* [40] (normally serves as upper bound for delivery probability), *PROPHET* v1 [23] (first proposal considered by the Delay-Tolerant Networks research community), and *Spray and Wait* [36] (appearing as lower bound for latency) for being the most cited proposals (i.e., often used as benchmark for performance comparison) [25]. For the social-aware approaches, *Bubble Rap* [17], *dLife and dLifecomm* [28] were selected as representative of solutions considering social structures (i.e., communities), node popularity, and dynamic behavior of users, being *dLifeComm* a variation of *dLife* based on the formation of communities as *Bubble Rap*.

Some of the simulated proposals need to have some parameters set, thus we set: (i) *PROPHET* with aging of delivery predictability happening at every 30 s; (ii) *Spray and Wait*, being binary and with a number of spraying copy set to 10; (iii) *dLifeComm* and Bubble *Rap*, with K-Clique ($k = 5$, 700 s familiar threshold) and cumulative window algorithms for community formation and node centrality computation; and (iv) *dLife* and *dLifeComm*, with 24 daily samples. The choice for such values is among those which each proposal has presented good overall performance as specified in their respective original papers.

Table 2.1 summarizes the setup parameters. It is important to say that our goal is not to show which proposal is the best. Instead, we want the reader to understand the potential of social-aware opportunistic solutions and why they haven't shown to be a trend in the last years for opportunistic networks.

2.5.3 Results

Before presenting the experiments results, here are a few general observations regarding our findings. The average number of contacts per hour is of approximately 962 in the heterogeneous scenario and of 32 in the trace-based one. Additionally, contacts are more sporadic in the trace-based scenario than in the heterogeneous one,

Table 2.1 Simulation parameters

Parameters	Values	
Simulator	Opportunistic Network Environment (ONE)	
Routing proposals	Epidemic, PROPHET, Spray and Wait, Bubble Rap, dLife and dLifeComm	
Scenarios	Heterogeneous	Trace Cambridge
Simulation time (s)	1,036,800	1,000,000
# of nodes	150 (people/vehicles)	36 (people)
Mobility models	Working Day, Bus, Shortest Path Map Based	Human
Node interface	Wi-Fi (Rate: 11 Mbps/Range: 100 m)	Bluetooth
Node buffer	2 MB	
Message TTL	1, 2, 4 days, 1 and 3 weeks	
Message size	1–100 kB	
# of messages	6,000 (only 5,018 considered for the heterogeneous scenario)	
Spraying copies	L = 10 (Spray and Wait)	
K-Clique	k = 5 and familiar threshold = 700 s (Bubble Rap and dLifeComm)	
Daily samples	24 (dLife and dLifeComm)	

in which contact frequency is more homogeneous. We also observe that the average number of unique communities is higher in the heterogeneous scenario (~68) than in the trace-based scenario (~8.7). Furthermore, most of the created communities encompasses all the existing nodes (150 for the heterogeneous simulations, and 36 for trace), which means that independently of the level of contact homogeneity, nodes are still well connected.

2.5.3.1 Heterogeneous Scenario

Figure 2.6 presents the results for the average delivery probability. It is clear that *Epidemic* (normally seen as an upper bound for this performance metric) has the worst performance amongst the proposals. This is due to the available buffer space in each device (2 MB), which was reduced to represent a limited willingness of the user to carry information of behalf of others.

As we could see [25–27], authors tend to consider unlimited storage. However, while this may be true in scenarios comprising nodes for the serving purpose (e.g., [1]), normally a user may find him/herself in an opportunistic network formed on-the-fly and surrounded by other users sharing storage space according to the capabilities of their devices and mostly to their willingness in cooperating.

PROPHET and *Bubble Rap* are also affected by the limited buffer space per device and their delivery capability diminishes as TTL increases. Since messages are allowed longer in the network (i.e., being replicated), this consequently results in buffer exhaustion. We believe that the mobility heterogeneity found in the scenario also contributes to the decrease in the delivery probability.

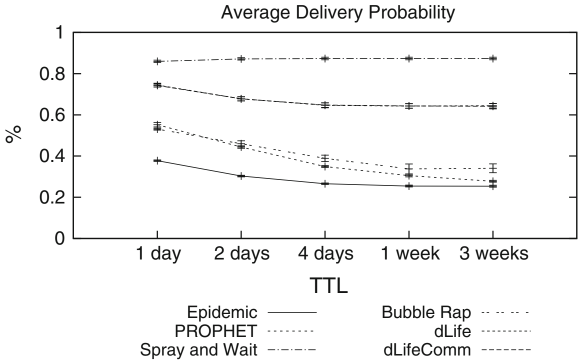


Fig. 2.6 Average delivery probability

Since users' dynamic behavior is considered, both *dLife* and *dLifeComm* are less affected by the limited buffer space. They carefully decide whether to replicate based on the social weight toward the destination and node importance in the network at the time of the encounter. Thus, wiser decisions take place as next forwarders are only chosen if they indeed have a stronger social connection or importance than the current message carriers.

Spray and Wait has the best performance. This is due to the fact that the scenario comprises nodes (e.g., buses and police patrols) that cover most of the simulated area and this proposal takes advantage of that. Most of its random replications happen to be to such nodes and as they move across the entire scenario added to the 100 m transmission range, its delivery capability increases. Despite the better performance, *Spray and Wait* still does not take advantage of longer TTLs to reach optimum delivery as some of the messages end up in the possession of nodes that are not well socially connected to their destinations.

Figure 2.7 shows the performance of each proposal considering the number of replicas created per delivered message. As one could expect, *Epidemic* and *Spray and Wait* are the upper and lower bounds for this metric. The former has the highest cost as it replicates a message with every encountered node that does not have a copy yet. The scenario itself is not epidemic-friendly since it has a high number of contacts among nodes. On the other hand, *Spray and Wait* is allowed to create up to 10 ($L = 10$) copies per messages, and thus its cost is the lowest (average across simulations of ~10.16 replicas per delivery). It is important to note that the way the ONE simulator determines cost (a.k.a. overhead) is given by the ratio between the number of successful forwarded message (discarding the forwarding to the final destination, i.e., number of delivered messages) and the number of delivered messages. This explains why cost here exceeds the one L specifies for *Spray and Wait*.

By being a probabilistic-based solution, PROPHET considers the frequency of past contacts with the destination of the message to decide on replication. Since the proposal ages such encounters, only nodes that have frequent contacts with the

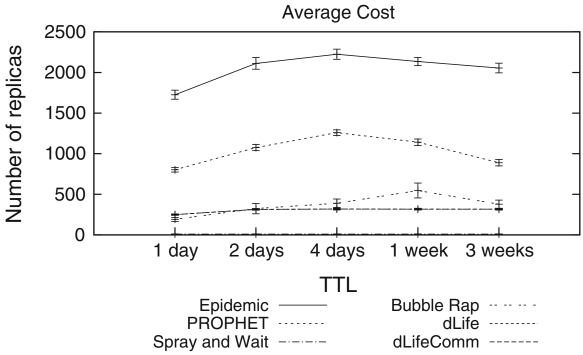


Fig. 2.7 Average cost

destination will be entitled to receive a copy. Despite the effort of the proposal, it still requires a high number of replicas to perform a successful delivery.

Bubble Rap, *dLife*, and *dLifeComm* have a much lower cost when compared to *Epidemic* and *PROPHET*, which shows that taking forwarding decisions based on some level of social similarity is beneficial as messages are replicated to nodes well socially connected to the destination of message. Regarding *Bubble Rap*, its cost is expected to increase as TTL increases [16, 17]: messages remain longer in the system and the solution takes advantage of this by building more reliable communities and refining centrality values to perform wiser decisions; however, there is still a drop in cost observed for a 3-week TTL. We believe this is due to the fact that the proposal has taken advantage of the longer TTL, choosing next forwarders much more wisely, which reduces its cost. As a matter of fact, this can be confirmed in Fig. 2.6 where the same performance is achieved, but with much less replications.

Both *dLife* and *dLifeComm* present a much more stable behavior as they consider the dynamism found in the users' daily routines and can choose to replicate only to nodes that actually have the best social interactions with the destination at each encounter.

Figure 2.8 shows the performance of the proposals in terms of average latency for delivering messages. As mentioned earlier, *Spray and Wait* takes advantage of the scenario which has nodes (i.e., buses and police patrols) covering the whole extension of the scenario and with longer transmission range. This significantly reduces its latency, as destinations can be reached in shorter periods of time as reported in its original paper [36]. *Epidemic* also has short latencies as spreading many copies of the same message also reduces the time it will take to reach its destination.

As waiting for the best (i.e., socially speaking) next forwarder may take some time, *dLife* and *dLifeComm* have a slight increase (varying between 490 and 4,350 s) in latency when compared to both *Epidemic* and *Spray and Wait*. Thus, one can observe the tradeoff when only forwarding messages in the presence of strong social links or highly important nodes in the current daily sample. Still, despite of taking

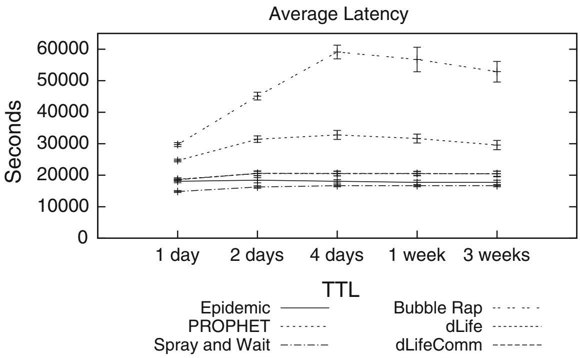


Fig. 2.8 Average latency

a little longer to reach their destinations, *dLife* and *dLifeComm* are able to deliver a reasonable amount of messages.

PROPHET does not consider the social strength that the next forwarder has with the destination; instead, it looks at the frequency of past interactions, resulting in replication of messages that may take longer to reach their destinations. This can also be confirmed with the performance of *Bubble Rap* regarding this metric: it chooses also to replicate according to the centrality (i.e., cumulative number of unique contacts within 6-h time windows) of nodes. Centrality indeed identifies nodes which may be well connected throughout the entire experiments, but fails to capture the different levels of "being socially connected" during the different periods of the day. To make matters worst given the number of average contacts per hour in this scenario (962), this proposal relies solely on global centrality while communities are forming. The result is replicas being created to nodes that have weak social ties with the destinations, which accounts for the total average latency experienced to deliver messages.

After looking at the performance of the proposals in a scenario of synthetic mobility models, we next present the performance of the same proposals using real human traces. With this, we expect to have experiments that are as close as possible from the reality when it comes to human mobility.

2.5.3.2 Cambridge Scenario

Figure 2.9 presents the performance of the considered proposal in terms of average delivery probability. One can easily see that the advantage of *Spray and Wait* seen in the previous scenario has decreased. This is mainly due to the fact that in this scenario there are no nodes covering the whole area. Instead, nodes are encountering others as they move throughout their daily routines. Additionally, contacts are much more sporadic. Under these circumstances, *Spray and Wait* spreads copies to nodes that may never come in contact with the destination, thus reducing its delivery capability.

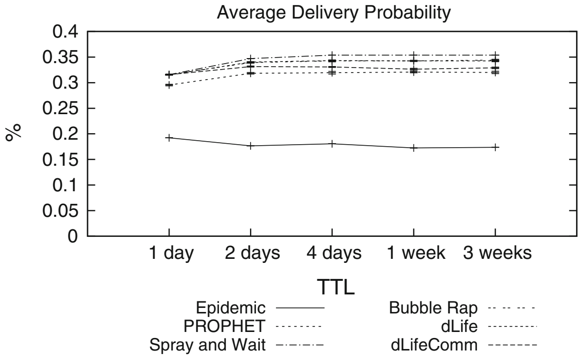


Fig. 2.9 Average delivery probability

This contact sporadicity also affects the remaining proposals since the solutions somehow depend on how nodes interact among themselves. Creating communities and determining centrality and importance of nodes as well as social weight among them take much longer in a scenario with low number of contacts (32 per hour). This is why *dLife* and *Bubble Rap* are statistically equivalent and perform very much at the same level. *dLifeComm* relies on the node importance to replicate, but since node importance takes longer to distinguish really important nodes, message are replicated to nodes not well socially related with the destination. Despite being based in the notion of community formation, computation of centrality in *Bubble Rap* is simpler (centrality is given by the the node's average degree over a set of time windows) and so can distinguish nodes much easier.

PROPHET faces the same issue, delivery predictability of nodes is not well computed and nodes getting copies are not the best option to increase delivery capacity. And despite of the uncontrolled replication, *Epidemic* does not reach its best in this scenario as: (i) nodes meet occasionally; and (ii) buffers have limited capacity.

In Fig. 2.10 we can see also the effect of the sporadic contacts. The number of copies to perform a delivery is much less since there are only few nodes to receive such copies at the time of exchange. Still *Epidemic* is the proposal with more replicas (despite of a decrease varying between 877 and 1,300 replicas per delivery) and *Spray and Wait* remains the proposal with the least cost, as one could already expect since its *L* limits the created number of messages' copies; however, this time with a little increase (average across simulations of ~15.12 replicas per delivery).

PROPHET experienced reductions varying between 252 and 606 replicas per delivery and remains as the second proposal to consume more resources. Social-aware solutions, namely *Bubble Rap*, *dLife* and *dLifeComm*, managed to have an average across simulations of approximately 24.52, 24.56, and 28.79 replicas per delivery. Here, we can observe the potential of social-aware opportunistic routing as with a few extra copies they can almost reach the same delivery of the social-oblivious *Spray and Wait* proposal.

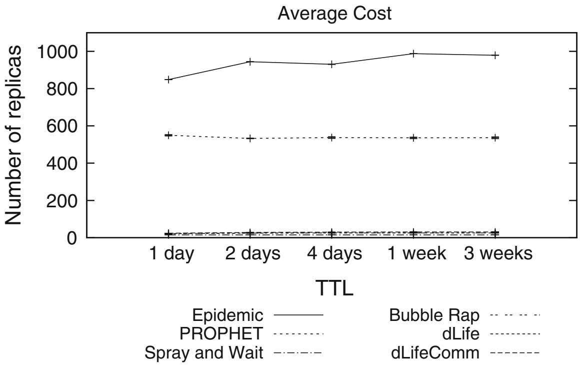
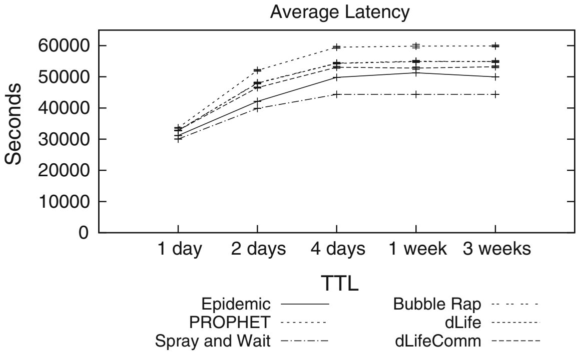


Fig. 2.10 Average cost

Fig. 2.11 Average latency



As contacts are rather sporadic, one could expect an increase in latency when delivering messages in the trace-based scenario. Indeed, this can be observed in Fig. 2.11 where all the proposals took longer time to deliver messages when compared to the heterogeneous scenario. To influence even more in the time to deliver messages, we observe an increase in the distance (i.e., average number of hops of at most 2) to reach the destination in the trace-based simulations. This means that, besides the time taken to decide whether or not to replicate, delivery also accounts for the time messages traverse different hops. The consequence is that for all cases, proposals almost doubled their time to deliver messages.

We believe this is due to the fact that most of the nodes meet each other (although in a sporadic manner) through the simulations, which means that proposals do not have information reliable enough to suitably decide on replication, resulting in an increase of the distance to reach destinations.

Amongst the proposals, *Bubble Rap* was the proposal that least experienced an increase in latency, having an almost similar behavior to the one observed in the heterogeneous scenario. As a matter of fact, this proposal experienced a decrease in latency with TTL of 4 days (less 4,709 s) and 1 week (less 1,725 s). This scenario has not been favorable to *PROPHET* (with the highest latency values), as contact sporadicity really affects the notion of time of last encounter employed by the proposal. *dLife* and *dLifeComm* were affected in the sense that the scenario is not dynamic given the nonhomogeneous frequency of contacts among nodes. Since it spreads many copies, *Epidemic* still keeps its position of second best solution in terms of latency as one of these many replicas will reach the destination in short times. And *Spray and Wait* has the lowest overall latency when it comes to delivering messages.

2.5.3.3 Considerations

A subset of social-oblivious and social-aware solutions were evaluated under the same conditions and considering two different scenarios. Each of these scenarios have particularities and specific challenges. In the first, heterogeneous scenario, nodes follow different mobility models (representing people, buses, and police patrols) and interact with each other as they move between home and work, go to a leisure activity, transport people, and secure the city. This is a scenario with a high number of contacts among nodes where social similarity can be inferred as they move throughout their daily routines.

We could observe that this scenario is advantageous to the social-oblivious *Spray and Wait* as it comprises nodes that cover most of the simulated area, nodes that get randomly spread copies. This way *Spray and Wait* manages to achieve a fast delivery of the highest number of messages (~86 %) with the lowest cost (at most 10.12 copies to perform a delivery). Still, the social-aware solutions, *dLife* and *dLifeComm*, are able to capture the dynamicity of user behavior and with a small latency tradeoff (25 % increase when compared to *Spray and Wait*) they reach up to 74 % delivery with a wise usage of resources (i.e., buffer space) producing at most 319 replicas to perform a successful delivery.

Since user dynamicity is not considered, *Bubble Rap* suffers from the problem of forming communities: as they are not readily available, the proposal solely relies on centrality to replicate information, which also takes some time to represent the system's reality. This results in a delivery probability not higher than 53 % with cost reaching up to ~547 replicas and latency increasing up to 250 % (in relation to *Spray and Wait*).

PROPHET takes into account the frequency of past interactions to decide on replication. However, having a high number of past interactions does not mean that nodes have enough time and suitable conditions to exchange information when they meet again. For instance, a stationary node may be in the path of a bus which results in a high frequency of past contacts; still, speed and physical obstacles, such as office walls, may lead to a problematic link, which in any case is considered a good link by PROPHET (i.e., high delivery predictability). Thus, unwanted copies (up to ~1,260 per delivery) are generated to reach just a little over 55 % delivery probability and that take up to 96 % (compared to *Spray and Wait*) more time to reach destinations.

Epidemic, considered as an upper bound to delivery probability, shows a different behavior: delivery probability up to ~38 % with the highest cost per delivered message (up to ~2,225 copies) but with lower latency (up to ~22 % compared to *Spray and Wait*). This is due to the fact that the proposal does not worry about resource constraints, and this scenario is also challenging given the small amount of buffer space nodes are willing to share (i.e., 2 MB).

As for the second scenario, trace based, besides the buffer limitations, it is even more challenging since contacts are sporadic and much less (32) when compared to the heterogeneous scenario (962). This consequently increases the time proposals need to have a suitable view of the network (in terms of the forwarding metrics they consider), increasing the time to decide on replication and to deliver messages.

The advantage of *Spray and Wait* is reduced in this scenario where it achieves up to 35 % of delivery with an increased cost of 15.37 copies per message delivered, while latency increases up to 44,372 s.

All the social-aware proposals keep a very close behavior to one another: reaching up to 33 % (*dLifeComm*) and 34 % (*dLife* and *Bubble Rap*) delivery predictability with a cost per delivery up to ~31 and 25 copies and latency increasing up to ~20 % (*dLifeComm*) and ~24 % (*dLife* and *Bubble Rap*) when compared to *Spray and Wait*.

Epidemic and *PROPHET* suffer even more with the sporadicity of contacts in this scenario, reaching up to 19 % and 32 % for delivery probabilities with an associated cost up to 987 and 536 copies per successful delivery, and latency increasing up to ~16 % and ~31 % when compared to *Spray and Wait*, respectively.

With these results one can conclude that still there is much work to be done in the sense of creating a new opportunistic routing solution, be it based on social-oblivious or social-aware approaches. What is more, social-aware solutions show a great potential in improving forwarding in opportunistic networks. It is clear that there is a tradeoff between delivery probability, cost and latency, since these proposals take more time to choose the best next hop. It is important to note that the performance of social-oblivious solutions such as *Spray and Wait* still require further investigation, because of its strong requirements. For instance, the number of spraying copies L during our experiments was set statically. However, according to its paper [36], this parameter must be computed based on the number of nodes in the network, and the current implementation used in our experiments does not take into account the effect of this.

Additionally, none of the proposals consider (multi)point-to-multipoint communication, which is a feature that should be looked upon given the need to efficiently spread content in opportunistic networks.

As mentioned before, our goal is not to elect the best solution, but instead to show the pros and cons of each of them, and most importantly to show that social-aware solutions deserve attention when it comes to developing routing approaches for opportunistic networks.

2.6 Conclusions

Social similarity has gained attention in the last years in the context of opportunistic networks given its potential to improve data forwarding. The reason behind this is that devices are carried by humans who happen to have distinct behaviors, which may help to identify one or many of them. Such behaviors can be related to social ties, work affiliation, shared interests, and among others, which can be used to infer/find social similarity. In addition, routing based on social similarity has proven to be much more stable that those based on mobility update (it is less volatile).

Thus, this chapter aims at introducing a new trend observed since 2007 with the appearance of social-aware opportunistic routing. For that, we cover a 12-year period worth of opportunistic routing solutions starting with a close look in the social-oblivious one to help us in understanding the need for this new trend. Then, we performed a close look at pioneer social-aware solutions along with new ones, covering how they use social similarity to devise their forwarding approaches.

The chapter also provides a brief look at the existing opportunistic routing taxonomies, in order to show when the social trend appeared and how much importance is given to it. We also take this opportunity to propose an update to an existing taxonomy, which includes the appearance of a new sub-category of social similarity solutions based on user dynamic behavior.

Finally, the chapter includes a set of experiments in different scenarios pointing out advantages and disadvantages of each social-oblivious and social-aware proposal, and indeed showing the potential of social-aware solutions to support our case: show that the trend is strong and deserves careful attention from researchers.

Acknowledgments Thanks are due to FCT for supporting the UCR (PTDC/EEA-TEL/103637/2008) project and Mr. Moreira's Ph.D grant (SFRH/BD/62761/2009), and to the colleagues of the DTN-Amazon project for the fruitful discussions.

References

1. Balasubramanian A, Levine B, Venkataramani A (2007) DTN routing as a resource allocation problem. In: Proceedings of the 2007 conference on applications, technologies, architectures, and protocols for computer communications, SIGCOMM '07. ACM, New York, pp 373–384. doi:10.1145/1282380.1282422. http://doi.acm.org/10.1145/1282380.1282422
2. Basagni S, Chlamtac I, Syrotiuk VR, Woodward BA (1998) A distance routing effect algorithm for mobility (dream). In: Proceedings of the 4th annual ACM/IEEE international conference on mobile computing and networking, MobiCom '98. ACM, New York, pp 76–84. doi:10.1145/288235.288254. http://doi.acm.org/10.1145/288235.288254
3. Bloom BH (1970) Space/time trade-offs in hash coding with allowable errors. Commun ACM 13(7):422–426. doi:10.1145/362686.362692. http://doi.acm.org/10.1145/362686.362692
4. Burgess J, Gallagher B, Jensen D, Levine BN (2006) Maxprop: routing for vehicle-based disruption-tolerant networks. In: INFOCOM 2006. 25th IEEE international conference on computer communications. Proceedings, pp 1–11. doi:10.1109/INFOCOM.2006.228
5. Cerf V, Burleigh S, Hooke A, Torgerson L, Durst R, Scott K, Fall K, Weiss H (2007) Delay tolerant network architecture. IETF Network Working Group. RFC 4838

6. Chaintreau A, Hui P, Crowcroft J, Diot C, Gass R, Scott J (2007) Impact of human mobility on opportunistic forwarding algorithms. IEEE Trans Mobile Comput 6(6):606–620. doi:10.1109/TMC.2007.1060. http://dx.doi.org/10.1109/TMC.2007.1060
7. Costa P, Mascolo C, Musolesi M, Picco GP (2008) Socially-aware routing for publish-subscribe in delay-tolerant mobile ad hoc networks. IEEE J Sel A Commun 26(5):748–760. doi:10.1109/JSAC.2008.080602. http://dx.doi.org/10.1109/JSAC.2008.080602
8. Daly EM, Haahr M (2007) 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. ACM, New York, pp 32–40. doI:10.1145/1288107.1288113. http://doi.acm.org/10.1145/1288107.1288113
9. D'Souza RJ, Jose J (2010) Routing approaches in delay tolerant networks: a survey. Int J Comput Appl 1(17):8–14 (Foundation of Computer Science)
10. Dubois-Ferriere H, Grossglauser M, Vetterli M (2003) Age matters: efficient route discovery in mobile ad hoc networks using encounter ages. In: Proceedings of the 4th ACM international symposium on mobile ad hoc networking & computing, MobiHoc '03. ACM, New York, pp 257–266. doi:10.1145/778415.778446. http://doi.acm.org/10.1145/778415.778446
11. Grasic S, Davies E, Lindgren A, Doria A (2011) The evolution of a DTN routing protocol—prophetv2. In: Proceedings of the 6th ACM workshop on challenged networks, CHANTS '11. ACM, New York, pp 27–30. doi:10.1145/2030652.2030661. http://doi.acm.org/10.1145/2030652.2030661
12. Grossglauser M, Vetterli M (2006) Locating mobile nodes with ease: learning efficient routes from encounter histories alone. IEEE/ACM Trans Netw 14(3):457–469. doi:10.1109/TNET.2006.876204. http://dx.doi.org/10.1109/TNET.2006.876204
13. Hossmann T, Spyropoulos T, Legendre F (2010) Know thy neighbor: towards optimal mapping of contacts to social graphs for DTN routing. In: Proceedings of the 29th conference on information communications, INFOCOM'10. IEEE Press, Piscataway, pp 866–874. http://dl.acm.org/citation.cfm?id=1833515.1833658
14. Hui P (2008) People are the network: experimental design and evaluation of social-based forwarding algorithms. Technical report UCAM-CL-TR-713, University of Cambridge, Computer Laboratory
15. Hui P, Crowcroft J (2007) How small labels create big improvements. In: Pervasive computing and communications workshops, 2007. PerCom workshops '07. Fifth annual IEEE international conference on, pp 65–70. doi:10.1109/PERCOMW.2007.55
16. Hui P, Crowcroft J, Yoneki E (2008) 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. ACM, New York, pp 241–250. doi:10.1145/1374618.1374652. http://doi.acm.org/10.1145/1374618.1374652
17. Hui P, Crowcroft J, Yoneki E (2011) Bubble rap: social-based forwarding in delay-tolerant networks. IEEE Trans Mobile Comput 10(11):1576–1589. doi:10.1109/TMC.2010.246. http://dx.doi.org/10.1109/TMC.2010.246
18. Jaho E, Karaliopoulos M, Stavrakakis I (2010) Social similarity as a driver for selfish, cooperative and altruistic behavior. In: Proceedings of the 2010 IEEE international symposium on a world of wireless, mobile and multimedia networks (WoWMoM), WOWMOM '10. IEEE Computer Society, Washington, DC, pp 1–6. doi:10.1109/WOWMOM.2010.5534930. http://dx.doi.org/10.1109/WOWMOM.2010.5534930
19. Jain S, Fall K, Patra R (2004) Routing in a delay tolerant network. In: Proceedings of the 2004 conference on applications, technologies, architectures, and protocols for computer communications, SIGCOMM '04. ACM, New York, pp 145–158. doi:10.1145/1015467.1015484. http://doi.acm.org/10.1145/1015467.1015484
20. Jones EPC, Li L, Ward PAS (2005) Practical routing in delay-tolerant networks. In: Proceedings of the 2005 ACM SIGCOMM workshop on delay-tolerant networking, WDTN '05. ACM, New York, pp 237–243. doi:10.1145/1080139.1080141. http://doi.acm.org/10.1145/1080139.1080141

21. Karamshuk D, Boldrini C, Conti M, Passarella A (2012) An arrival-based framework for human mobility modeling. In: World of wireless, mobile and multimedia networks (WoWMoM), 2012 IEEE international symposium on a, pp 1–9. doi:10.1109/WoWMoM.2012.6263683
22. Keränen A, Ott J, Kärkkäinen T (2009) The one simulator for DTN protocol evaluation. In: Proceedings of the 2nd international conference on simulation tools and techniques, Simutools '09. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, pp 55:1–55:10. doi:10.4108/ICST.SIMUTOOLS2009.5674. http://dx.doi.org/10.4108/ICST.SIMUTOOLS2009.5674
23. Lindgren A, Doria A, Schelén O (2003) Probabilistic routing in intermittently connected networks. SIGMOBILE Mob Comput Commun Rev 7(3):19–20. doi:10.1145/961268.961272. http://doi.acm.org/10.1145/961268.961272
24. Liu C, Wu J (2009) An optimal probabilistic forwarding protocol in delay tolerant networks. In: Proceedings of the tenth ACM international symposium on mobile ad hoc networking and computing, MobiHoc '09. ACM, New York, pp 105–114. doi:10.1145/1530748.1530763. http://doi.acm.org/10.1145/1530748.1530763
25. Moreira W, Mendes P (2011) Survey on opportunistic routing for delay/disruption tolerant networks. Technical report SITI-TR-11-02, Research Unit in Informatics Systems and Technologies (SITI), University Lusofona
26. Moreira W, Mendes P, Sargento S (2011) Assessment model for opportunistic routing. In: Communications (LATINCOM), 2011 IEEE Latin-American conference on, pp 1–6. doi: 10.1109/LatinCOM.2011.6107393
27. Moreira W, Mendes P, Sargento S (2012) Assessment model for opportunistic routing. Lat Am Trans IEEE (Rev IEEE Am Lat) 10(3):1785–1790. doi:10.1109/TLA.2012.6222585
28. Moreira W, Mendes P, Sargento S (2012) Opportunistic routing based on daily routines. In: World of wireless, mobile and multimedia networks (WoWMoM), 2012 IEEE international symposium on a, pp 1–6. doi:10.1109/WoWMoM.2012.6263749
29. Moreira W, de Souza M, Mendes P, Sargento S (2012) Study on the effect of network dynamics on opportunistic routing. In: Proceedings of the 11th international conference on ad-hoc, mobile, and wireless networks, ADHOC-NOW '12. Springer, Berlin, pp 98–111. doi:10.1007/978-3-642-31638-8_8. http://dx.doi.org/10.1007/978-3-642-31638-8_8
30. Mtibaa A, May M, Diot C, Ammar M (2010) Peoplerank: social opportunistic forwarding. In: Proceedings of the 29th conference on information communications, INFOCOM '10. IEEE Press, Piscataway, pp 111–115. http://dl.acm.org/citation.cfm?id=1833515.1833538
31. Nelson S, Bakht M, Kravets R (2009) Encounter-based routing in DTNs. In: INFOCOM 2009, IEEE, pp 846–854 . doi:10.1109/INFCOM.2009.5061994
32. Nguyen HA, Giordano S (2012) Context information prediction for social-based routing in opportunistic networks. Ad Hoc Netw 10(8):1557–1569. doi:10.1016/j.adhoc.2011.05.007. http://dx.doi.org/10.1016/j.adhoc.2011.05.007
33. Ramanathan R, Hansen R, Basu P, Rosales-Hain R, Krishnan R (2007) Prioritized epidemic routing for opportunistic networks. In: Proceedings of the 1st international MobiSys workshop on mobile opportunistic networking, MobiOpp '07. ACM, New York, pp 62–66. doi:10.1145/1247694.1247707. http://doi.acm.org/10.1145/1247694.1247707
34. Scott J, Gass R, Crowcroft J, Hui P, Diot C, Chaintreau A (2006) CRAWDAD trace cambridge/haggle/imote/content (v. 2006-09-15). http://crawdad.cs.dartmouth.edu/cambridge/haggle/imote/content
35. Song L, Kotz DF (2007) Evaluating opportunistic routing protocols with large realistic contact traces. In: Proceedings of the second ACM workshop on challenged networks, CHANTS '07. ACM, New York, pp 35–42. doi:10.1145/1287791.1287799. http://doi.acm.org/10.1145/1287791.1287799
36. Spyropoulos T, Psounis K, Raghavendra CS (2005) 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. ACM, New York, pp 252–259. doi: 10.1145/1080139.1080143. http://doi.acm.org/10.1145/1080139.1080143

37. Spyropoulos T, Psounis K, Raghavendra CS (2007) Spray and focus: efficient mobility-assisted routing for heterogeneous and correlated mobility. In: Proceedings of the fifth IEEE international conference on pervasive computing and communications workshops, PERCOMW '07. IEEE Computer Society, Washington, DC, pp 79–85. doi:10.1109/PERCOMW.2007.108. http://dx.doi.org/10.1109/PERCOMW.2007.108
38. Spyropoulos T, Psounis K, Raghavendra CS (2008) Efficient routing in intermittently connected mobile networks: the single-copy case. IEEE/ACM Trans Netw 16(1):63–76. doi:10.1109/TNET.2007.897962. http://dx.doi.org/10.1109/TNET.2007.897962
39. Spyropoulos T, Rais RN, Turletti T, Obraczka K, Vasilakos A (2010) Routing for disruption tolerant networks: taxonomy and design. Wirel Netw 16(8):2349–2370. doi:10.1007/s11276-010-0276-9. http://dx.doi.org/10.1007/s11276-010-0276-9
40. Vahdat A, Becker D (2000) Epidemic routing for partially connected ad hoc networks. Technical report CS-200006, Duke University
41. Zhang Z (2006) Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges. Commun Surv Tutor 8(1):24–37. doi:10.1109/COMST.2006.323440. http://dx.doi.org/10.1109/COMST.2006.323440
42. Zhao W, Ammar M, Zegura E (2005) Multicasting in delay tolerant networks: semantic models and routing algorithms. In: Proceedings of the 2005 ACM SIGCOMM workshop on delay-tolerant networking, WDTN '05. ACM, New York, pp 268–275. doi:10.1145/1080139.1080145. http://doi.acm.org/10.1145/1080139.1080145

Chapter 3
Context-Based Routing Protocols for OppNets

Anshul Verma, K. K. Pattanaik and Aniket Ingavale

Abstract Opportunistic network does not rely on any pre-existing infrastructure and no assumption is made on the existence of a complete path between source and destination. The development of efficient routing protocols for opportunistic networks is generally a difficult task due to the absence of knowledge about the network topology. Therefore, routing is one of the most persuasive challenges of opportunistic networks. In this chapter, we give the general definition of context information and emphasize the role of context information to take forwarding decisions in opportunistic networks. We classify the main routing protocols proposed in the literature on the basis of context information of users they exploit. Specifically, we classify two main classes of routing protocols, corresponding to context-oblivious and context-aware protocols. Then, we further classify context-aware routing protocols into two subclasses, i.e., partially context-aware and fully context-aware protocols, based on the amount of context information they use for routing. We emphasize and describe only routing protocols of context-aware category and the detailed discussion about context-oblivious routing protocols is out of the scope of this chapter.

Keywords Opportunistic network · Routing protocol · Context information · Context aware routing

3.1 Introduction

The opportunistic network is an extension of Mobile Ad hoc Network (MANET). Several properties of traditional wireless networks, such as disconnection of nodes, mobility of users, network partitions, and links' instability, are treated as drawbacks. This makes the design of MANET routing protocols significantly more difficult [1].

A. Verma (✉) · K. K. Pattanaik · A. Ingavale
Department of Computer Science, ABV-Indian Institute of Information Technology
and Management, Block: A, Room No.: 117, Gwalior 474015, Madhya Pradesh, India
e-mail: anshulverma87@gmail.com

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_3,

Opportunistic networks [2] are formed out of portable mobile devices carried by people and does not rely on any pre-existing network infrastructure. Opportunistic networks consider disconnections, partitions, mobility etc., as features instead of the drawbacks. The communication between disconnected groups of nodes is achieved through the mobility of nodes. In opportunistic network, a complete path between source and destination does not presume, because they might never be connected to the same network at the same time [3]. It is achieved by removing the assumption of physical end-to-end connectivity and allows such nodes to transmit messages by using the store-carry-and-forward approach [4]. In which intermediate nodes are used to store messages during the unavailability of forwarding opportunity toward the destination, and any future contact opportunity with other mobile devices is exploited to bring the messages more closely to the destination. The performance of routing protocol depends on the knowledge about network topology, absence of such knowledge makes the designing of routing protocols more complicated [5]. Unfortunately, this kind of information cannot easily be disseminated in opportunistic networks.

Context information plays a significant role in designing efficient routing protocols. Context information represents the current working environment and behavior of users, such as users' working address and institution, the probability of visiting particular places or meeting with other users etc. Routing protocols can effortlessly identify suitable forwarders based on the context information about the destination [6]. The context information is basically used to make routing protocols capable to learn the network state, automatically adjust to its dynamic nature, and thus improve their operations [7, 8]. In this chapter, we classify the main existing routing protocols proposed in the literature based on the amount of context information they exploit. Specifically, we identify two main classes: context-oblivious and context-aware protocols. Furthermore, we classify context-aware category into two sub-categories: partially context-aware and fully context-aware protocols.

The rest of this chapter is organized as follows. In Sect. 3.2 we give a general definition of the term context information for any context-sensitive network communication application. Section 3.3 describes routing concept of opportunistic networks. In Sect. 3.4 we represent a short discussion of routing protocols of context-oblivious family. We illustrate a detailed discussion of most popular routing protocols of context-aware category in Sects. 3.5 and 3.6. We summarize and conclude in Sect. 3.7.

3.2 Context Information

With the evolution of pervasive and ubiquitous computing, new techniques to compute or provide services are emerging. The main advantage of these new techniques is that they do not provide same services for each user in the same way. Rather, they provide services on the basis of interests, preferences or abilities of single users, or specific user groups. Therefore, they are known as context-aware services. To make utilization of these services the context information of a user, of his device and of his environment must to be described, collected, and analyzed properly. But before utilization of context information, its meaning must properly be defined, which is still interpreted differently in the literature according to its application. A general definition of context information based on the widespread literature study, which suits to all areas of life, is given below.

Different classifications and views about context information are given here. First of all, we can classify context information into three main categories: user context, terminal context, and communication network context. For different applications or services, these classes can be combined or interpreted as required.

In [9], context is described as identities of neighbors around the user, position, time of day, temperature, and season. However, these are more special context information those were not included in general definition. The definition in [10] is similarly carried out but temperature and season are not considered as context there. In [11], context is defined as applications, environment, status, surroundings, and situation. Furthermore, in the authors' opinion the definition given by A. Dey and G. Abowd is the widely acceptable. Because, their definition describes the complete range of context information in a common way that it can be used in mostly real-life applications. The definition is given below [12]:

"Context is any relevant information that describes the environment of several objects (i.e. a person, object, or location) that can be used to the interaction between an application and a user. Context may be place, person, groups, and physical and computational objects."

Three different entities people, places, and things were identified in [13]. The entity places describe geographical spaces like homes, schools, buildings, and so on. People were categorized in groups or individuals. The term things were identified as physical objects or software components. These entities can be described by four categories:

- **Identity**—each entity of the application is characterized by a unique identifier.
- **Location**—includes location related information of identities as positioning data and orientation as well as information about regional relations to neighboring entities.
- **Status**—represents nature, condition, and environment of entities. For example, the status of a place can be described as the current temperature, the weather condition or the noise level.
- **Time**—refers to both date and time.

3.3 Routing in Opportunistic Networks

In opportunistic network there is no need to establish a complete path from source to destination. This phenomenon drastically reduces the complexity of routing protocols in opportunistic networks. But, new challenges of routing come into pictures that are distinct from traditional networks' challenges. The routing protocols of opportunistic networks are capable to transmit data with some reliability even when the disconnection is frequently occurred in the network or establishment of end to end is not possible. Moreover, since communication between nodes may occur randomly without any prior information, other routing approaches of conventional wireless network can't be applied. In such cases, flooding based routing protocols are very popular to provide communication. However, this approach generates extra traffic overhead and increases energy consumption.

The performance of the routing protocol improves when context information of the network is exploited, i.e., knowledge about network topology, users' behavior, and information about the users themselves etc. Each context-aware routing protocol exploits all or some specific type of context information according to the design specification of routing. The context information could be the home address, office, school, profession, phone number, the mobility pattern of users, the frequency to visit a particular place, or the communities that the users belong to, and so on. All these information are helpful in taking decision to transmit messages. For example, to find out a best forwarder for communication toward the final destination, the home address of intermediate node is a precious piece of context information.

In the following section, we classify the main existing routing protocols into two main classes: context oblivious and context aware, on the basis of context information they exploit as given in [14]. Then we further classify context-aware routing protocols into two subclasses: partially context-aware and fully context-aware protocols, based on the amount of context information they use for routing. Specifically, we provide detailed description of context-aware routing protocols and emphasize the role of context information.The discussion about context-oblivious routing protocols is out of the scope of this chapter. Along with it, we also provide an extremely quantitative comparison between representatives of context-oblivious and context-aware protocols.

3.4 Context-Oblivious Routing

Context-oblivious routing category contains flooding-based routing protocols. These routing protocols either follow blindly flooding or controlled flooding techniques. The routing protocols of this category are only solution when information, about possible path toward the destination or suitable next hope node, is not available. It means they do not exploit any form of context information. The message is transmitted to destination by disseminating as widely as possible.

In flooding-based techniques, a source node transmits packets to all its neighbors. Each neighbor receives a packet and forwards it to all its neighbors except the one from which the packet had arrived. The destination may receive each packet via different route at a different time. Flooding-based routing follows simplest route finding approach, and gives minimum delay ratio to establish a connection between two nodes, because it does not require any network topological information or any context information. However, flooding-based technique disseminates a huge number of packets in the network, which generates network congestion problem, and is also very costly in terms of memory and energy consumption [15]. One common approach to solve this problem is control flooding by limiting the maximum number of packets a network can have at a time, or by limiting the maximum number of intermediate nodes a packet can travel, or by setting Time to live (TTL) for each packet.

Epidemic routing is the most suitable protocol of this category [16]. In this routing protocol, each node maintains a summary vector that indicates compact representation of the messages currently store in the local buffer. There are several other routing protocols come under this context-oblivious category, e.g., Network coding [17], Hybrid routing [18], Spray and wait [19] etc.

3.5 Partially Context-Aware Routing

Partially context-aware routing protocols use some specific type of context information, on the basis of which they take forwarding decisions. The main difference between partially and fully context-aware routing protocols is that the former uses some particular type of context information, while the latter exploits and manages any type of context information.

3.5.1 PROPHET Routing Protocol

One of the most admired examples of this class is Probabilistic Routing Protocol using History of Encounters and Transitivity (PROPHET) [20]. It is an extension of epidemic routing protocol [16] with the concept of delivery probability. The delivery probability is the predictability for a node to the successful message delivery to a specific destination. In real life, users mostly move in a predictable manner according to their repeating behavioral patterns, e.g., if a node has visited a place many times before, there is probability that it will visit that place again. On the basis of these observations, the protocol uses frequency of meetings between nodes as context information to improve the routing performance.

According to the protocol, each node a maintains a probabilistic metric called delivery predictability, $P_{(a,\,b)} \in [0, 1]$, for each known destination b. This represents the delivery probability of this node to that destination. Like epidemic routing protocol, when two nodes come within the communication range of each other, they exchange summary vectors along with delivery predictability information maintained by these nodes. The protocol uses this information to update the internal delivery predictability as given below. Moreover, the summary vector information is used to request the desire messages from the other node according to forwarding strategy.

3.5.1.1 Delivery Predictability Calculation

The delivery predictabilities’ calculation is divided into three parts. First, whenever nodes meet with others they update their probabilistic metrics, so the nodes that are often meet have a high delivery probability. The Eq. (3.1) shows this calculation, where $P_{init} \in [0, 1]$ represents initialization constant.

$$P_{(a,\,b)} = P_{(a,\,b)old} + (1 - P_{(a,\,b)old}) \times P_{init} \tag{3.1}$$

If two nodes do not meet with each other within a particular time, they are not good forwarder of message to each other, so the delivery probability values must be reduced. The Eq. (3.2) represents aging equation, where $\gamma \in [0,1]$e represents aging constant, and k is the number of time units that have elapsed since the last time the nodes were meet. The time unit may be differ according to application and network requirements.

$$P_{(a,\,b)} = P_{(a,\,b)old} \times \gamma^{k} \tag{3.2}$$

The delivery predictability also follows transitive property, according to this if node A frequently meets node B, and node B frequently meets node C, then nodes A and C also have a good message delivery probability to each other. The transitivity affects the delivery predictability as shown in Eq. (3.3), where $\beta \in [0, 1]$ represents scaling constant which decides the impact of transitivity on the delivery predictability.

$$P_{(a,\,c)} = P_{(a,\,c)old} + (1 - P_{(a,\,c)old}) \times P_{(a,\,b)} \times P_{(b,\,c)} \times \beta \tag{3.3}$$

3.5.1.2 Forwarding Strategies

In traditional routing protocols, the selection of the next forwarder is a simple task; neighbor node that has the shortest path or lowest cost to the destination is selected. Normally, a rout from source to destination is find out and established before transmission, thus reliability of the rout is very high. However, the scenario is completely different here. When a node receives a message, there might not be any next forwarder so the node has to store the message in local buffer, and upon each meeting with another node, the decision must be taken whether or not to forward the message. Furthermore, the message should be forwarded to multiple nodes to increase the delivery probability of message.

We can select a static threshold value and only give a message to nodes that have grater delivery probability than the fixed threshold for the destination of the message. On the other hand, when meets with a node with low delivery probability, it is not sure that a node with higher delivery probability will meet within a sufficient time. Furthermore, the selection of adequate number of forwarders to a message is also difficult. PROPHET chooses a simple forwarding strategy, when two nodes come within the communication range of each other, a message is given to the other node if the delivery probability to the destination of the message is greater at the other node.

3.5.2 MV Forwarding Algorithm

The routing protocol MV [21] was designed for efficient message delivery in opportunistic network. It exploits the frequency of meetings between nodes and also exploits information about the frequency of visits to particular physical locations. The historical data are used to rank each message in a node's buffer according to the possibility of delivering a message. MV recognizes the nodes' motion patterns and exploits it to improve the performance of routing.

3.5.2.1 Assumption

MV supports only the type of networks that follow the following three assumptions.

1. Nodes have an unlimited size buffer for their own messages, and only a limited size buffer for the messages they receive from others.
2. When nodes have a chance for transfer, they transmit with a fully reliable and unlimited bandwidth link layer. MV breaks this limitation and isolates routing protocols independent of the limits of data link layer.
3. Massages are eventually transmitted to stable destinations located on geographic locations.

3.5.2.2 The MV Algorithm

The MV routing algorithm works as follows. When a node A encounters another node B, they exchange message through a number of steps. First, node A gives to B a list of the massages it contains along with their destinations. Each message is also explained by A with A's possibility of delivery according to the below given formula. Node B gives the same list to A and A calculates the possibility of delivering B's messages. Node A now sorts the merged lists by the possibility of delivery, deletes its own and all messages for which B has a higher possibility of delivering. Node A then chooses the top n messages remaining, and requests from B all the messages that are not already received.

This is the similar routing algorithm that the same researchers have published in their previous work [22], except the technique for determining what messages have most possibility to be delivered. MV decides a probability, $P_n^k(i)$, that the current node, k, can eventually deliver a message to a destination i within n attempts.

3.5.2.3 Probability of Delivery

The $\mathrm{P}_0^{\mathrm{k}}(\mathrm{i})$ describes the probability of transmitting a message to some node k in a single attempt. In this situation, the message delivery probability will be equal to the node's probability of visiting the destination area. MV assumes that the probability of visiting a area in future is highly correlated with the peer's history of visiting a region.

Accordingly, for each node *k*, MV has a vector P_0^k with one entry for each area. Each entry *i* of $\mathrm{P}_0^k(\mathrm{i})$ depends on the recorded movement of the node during the last *t* round, here a round is defined as a fixed length of time (e.g., 1 day of 1 h, depending on the movement speed of the node): $\mathrm{P}_0^k(\mathrm{i}) = \mathrm{t}_\mathrm{i}^k/\mathrm{t}$, here t_i^k represents number of times node *k* visited cell *i* during the previous t visits.

Second, MV assumes messages can be transmitted to maximum one node before being transmitted to the destination. Both the current node *k* and the intermediate node *j* have a copy of the message, and any one or both can deliver it.

Let $\mathrm{P}_1^k(\mathrm{i})$ is the probability of eventually delivering a message to area *i* starting with node *k* and by using maximum one intermediate node. This is described by:

$$\mathrm{P}_1^k(\mathrm{i}) = 1 - \prod_{\mathrm{j}=1}^{\mathrm{N}} (1 - \mathrm{m_{jk}} \mathrm{P}_0^\mathrm{j}(\mathrm{i})) \tag{3.4}$$

where N represents the number of nodes in the system, $\mathrm{m_{jk}}$ is the probability of node k and node j that are visiting the same area together. Like movement probability, MV has meeting probability on the basis of meetings during the last t visits: $\mathrm{m_{jk}} = \mathrm{t_{j,k}}/\mathrm{t}$, where $\mathrm{t_{j,k}}$ represents the number of times nodes j and k are in the same area. Note, $\mathrm{m_{jj}} = 1$. Equation (3.4) describes the probability that neither node k nor any other node k visits the destination directly. Finally, MV assumes that messages can be forwarded to no more than n other nodes:

$$\mathrm{P}_\mathrm{n}^k(\mathrm{i}) = 1 - \prod_{\mathrm{j}=1}^{\mathrm{N}} (1 - \mathrm{m_{jk}} \mathrm{P}_{\mathrm{n}-1}^\mathrm{j}(\mathrm{i})) \tag{3.5}$$

Equation (3.5) is unable to scale with the number of intermediate hops or nodes in the system. To compute the probability, the meeting patterns of all other nodes must be known. MV has found in evaluations that $\mathrm{P}_1^k(\mathrm{i})$ is approximate similar to $\mathrm{P}_\mathrm{n}^k(\mathrm{i})$ to serve.

3.5.3 The MaxProp Protocol

Here we describe the MaxProp protocol [23] and major assumptions made to describe it. Opportunistic network can operate in several environment, e.g., on pedestrians, vehicles, animals, or underwater sensors. Here, the protocol is creating an opportunistic network by using buses and desktop computing devices.

3.5.3.1 Model

The protocol assumes that each node has an unlimited size buffer to store its own messages, while a fixed size buffer is used to store messages generated by others. Duration and bandwidth are the constraints for the transfer opportunity.

The protocol assumes that nodes do not have any prior knowledge about environmental conditions, i.e., network connectivity, nodes' movement pattern, nodes' geographic location, etc.

Opportunistic network performs three major operations.

1. **Neighbor Discovery**: Nodes must discover their neighbor nodes before a transmission starts.
2. **Data Transfer**: When two nodes come within the communication range of each other, they can exchange the data. Nodes do not know the duration of meeting.
3. **Storage management**: Each node must maintain its fixed size buffer space by deleting stored packets according to some algorithm. Messages for that the receiving node is destination are passed up to the upper network layer and deleted from the buffer.

Each node carries stored messages until it meets with other nodes. A node will continuously forward a message to any number of encountered nodes until the message times out, the delivery of message is notified by an ack, or the message is dropped due to buffer overflow.

3.5.3.2 Protocol Definition

The MaxProp protocol uses various approaches to increase the delivery rate and reduce the latency of delivered packets. MaxProp exploits several approaches to describe the order of packets transmission and deletion these mechanisms are illustrated in Fig. 3.1. The main part of protocol is a ranked list of packets, stored in a node, according to cost assigned to each destination. The cost is an approximation of delivery possibility. Moreover, MaxProb uses acknowledgments to confirm packet delivery. MaxProb gives a higher priority to new packets, and also prevents reception of duplicate packets. The rest part of this section describes the mechanisms of destination cost estimation and buffer management.

Estimating Delivery Likelihood: In literature, it has been demonstrated that the optimal path for delivery in opportunistic network can be generated by creating a directed graph of nodes [24]. A variation of Dijkstra's algorithm is used to determine the optimum shortest path. In literature, no approaches are available to predict the future connections. Therefore, MaxProp assigns link weights according to following rules.

Let s is a set of nodes in the network. Each node i calculates and maintains a probability of meeting with node j, here i and j are $i \in s$ and $j \in s$. We calculate this probability f_j^i as the possibility that the next encountered node will be j. Initially, f_j^i set to $1/(|s| - 1)$ for all nodes. The value of f_j^i is incremented by 1 on each meeting with node j, and all values of f are also re-calculated. This method is called incremental averaging, the nodes that are encountered infrequently assigned lower values over time. These values are exchanged each time when two nodes meet.

For example, for an opportunistic network with four other nodes, a node j has values for $f_1^i = f_2^i = f_3^i = f_4^i = 0.25$. Upon meeting with node 3, the node

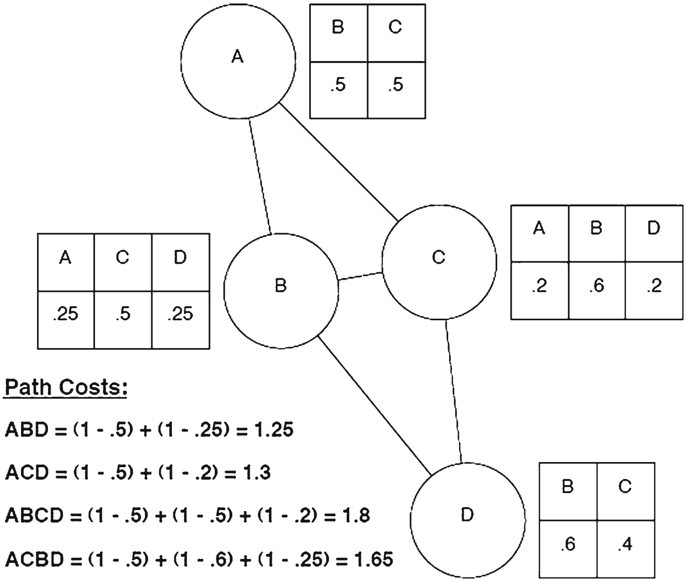


Fig. 3.1 MaxProp path cost calculation

updates $f_3^i = 1.25$ and re-calculates all values, so that the sum of them is 1 again: for $f_1^i = f_2^i = f_4^i = 0.125$ and $f_3^i = 0.625$.

With the help of other nodes' values, a local node computes a cost, $c(i, i+1, \ldots, d)$, for each possible path to the destination d by using up to n intermediate hops. The total cost for a path, using all intermediate connections from source to destination, is the sum of the probabilities that each connection on the path does not exist, calculated as one minus the probability that each link exists:

$$c(i, i+1, \ldots, d) = \sum_{x=1}^{d-1} [1 - (f_{x+1}^{x})] \tag{3.6}$$

The lowest path cost among all possible paths is selected as the cost for a destination. The example of this method is illustrated in Fig. 3.1, where the minimum value 1.25 is selected as cost from A to D.

In MaxProp, highest priority rank packets are transmitted first during a transfer opportunity. Lowest priority rank packets are deleted first to make space for new incoming packets. In the situation, when two packets have same cost to destinations, the packets that have traveled fewer intermediate nodes are allowed first to be transmitted.

Complementary Mechanisms: MaxProp uses several other techniques with this core to increase the delivery rate and reduce latency.

When two nodes come within the communication range of each other, they exchange packets in a specific priority order.

1. The messages that are destined to the neighbor node are transferred.
2. The routing information is shared between nodes; specifically, a vector containing the probability of meeting with other nodes.
3. The delivery of data is confirmed by sending acknowledge.
4. Packets that have not been forwarded for a long time are given priority. It is found in simulation that delivery possibility can appreciate packets that have a high possibility of reaching a destination due to this some packets never get a chance of being forwarded. Therefore, MaxProp gives higher priority to new packets to provide more chances of being forwarded. So newer packets are transmitted several times when get opportunity and increasing the chance of eventual delivery. MaxProp implements this strategy by splitting the buffer in two parts on the basis of threshold t hops. Packets above the threshold are sorted by the scoring mechanism explained above and packets below are sorted by hopcount. The adaptive approach to setting the threshold is explained in [23].
5. The remaining untransmitted packets are transmitted in a sequence based on the scores as explained in estimating delivery likelihood heading.
6. Finally, the packets that have already been transmitted to the node are not transmitted again. Each packet stores a hop list that represents nodes that the packet has already visited.

Managing Buffers: There are only three situations when a node n can drop a packet p, without affecting the overall delivery rate of the network.

1. A copy of packet p has already been delivered to its destination.
2. The route with adequate bandwidth will not exist between node n and destination during the life time of packet p.
3. No copy of packet p has been delivered, but some copy of packet p will be delivered even if node n deletes its copy.

These three situations are mutually exclusive, because it is not possible for any packet to meet more than one of them. Only the situation not covered is that packet p has not been delivered but can only be delivered if node n holds on to p. Therefore, deletion of this type of packet will affect the overall delivery rate. Since the propagation of information in opportunistic network is very slow, a node will normally not know the values of the above three situations with certainty.

To check whether situation 1 has been satisfied, Maxprop uses acknowledgments sent from the destination and disseminated to all nodes in the network. To check situation 2, MaxProp uses the scoring mechanism from estimating delivery likelihood heading. Situation 3 is the most critical to check. MaxProp uses hop count as a weak estimator. Because packets are copied from one node to another, packets that have disseminated further are assigned lower priority for routing and are deleted first.

In sum, MaxProp removes acknowledged packets instantly, followed by packets that have crossed the threshold of t intermediate hops with minimum scores, followed by packets with the maximum hops below threshold t.

3.5.4 MobySpace: A Mobility Pattern Space

In this routing protocol [25], the mobility pattern of nodes is used as context information to take routing decisions. The protocol uses a high dimensional Euclidean space, called MobySpace. Each axis of MobySpace represents the possibility of meeting between two nodes and the gap between them represents the probability of that meeting to occur. Two nodes that contact with similar set of nodes with similar frequencies are very close in the MobySpace. The node that is closer to the destination node is a best forwarder of message.

The packets are delivered by forwarding toward the nodes that have more similar mobility patterns as destination's mobility pattern. In MobySpace, the mobility pattern of node indicates its coordinates, named MobyPoint, eventually delivery is performed by giving packets to the nodes that have MobyPoints most closer to the MobyPoint of the destination.

The following section represents the ways to characterize the mobility patterns and the manners to manage these patterns by the nodes, and possible limits and issues are also represented.

3.5.4.1 Mobility Pattern Characterization

Mobility patterns are characterized on the basis of the number and the type of the dimension of the particular MobySpace. MobySpace does not represent the geographical location of the node, while it describes some features of a node's mobility pattern. There are several methods to describe the mobility pattern, each requires some basic requirements. Mobility patterns should be very simple to measure in order to reduce the computational cost and to reduce overhead associated with spreading them within the network. Moreover, they must be helpful to take efficient routing decisions. A mobility pattern can be constructed on the basis of historical information regarding meetings that the node has encountered in past. A recent study of such mobility patterns is represented in [25].

It highlights that if we want to transmit packet from one node to another, we should select a relay that frequently meets the destination. A MobySpace based on this type of pattern will work as follows. Each permissible contact is indicated by an axis, and the distance along that axis represents the probability of meeting. Two nodes that meet with similar set of nodes with similar frequency are very close in the space. While, nodes that meet with different set of nodes or meet with same set of nodes but with different frequencies have distance between each other.

3.5.4.2 Mobility Pattern Acquisition

A node can recognize its own mobility pattern by several methods. A node can observe its mobility pattern by learning its contacts or its frequency of visits to various places. The tags are fixed to each location, so that the nodes can identify its current position. Alternatively, nodes may be able to investigate an existing infrastructure to construct these patterns.

In the same manner, a node can obtain mobility patterns of other nodes by using several methods. First, the mobility patterns can be spread in dissemination-based approach. Second, nodes can spread only efficient information of mobility patterns to minimize the buffer and network resource consumption. Finally, nodes can store their mobility patterns at predefined storage locations, and they also update their information with the data stored at these storage locations.

3.5.4.3 Mobility Pattern Usage

As we mentioned earlier, coordinates of a node in MobySpace are determined by its mobility pattern, it is represented by the position of MobyPoint associated with it. The core idea is that packets are given to nodes having similar mobility pattern as of the destination. Let U and L respectively represent the set of all nodes and locations. The MobyPoint for a node $k \in U$ indicates a point in an n-dimensional space, here $n = |L|$. We use $mk = (c1k, \ldots, cnk)$ to represent the MobyPoint of node k. And $d(mi, mj)$ represents the distance between two MobyPoints i and j.

Directly connected neighbors of node k at any time t can be represented by $W_k(t) \subseteq U$. The group of neighbors that also contains k is represented by $W_k^+(t) = W_k(t) \cup \{k\}$. MobySpace routing chooses only one of these neighbors for receiving or keeping packets. The routing function f exploits the neighbor that is nearest to the destination b. The function f is used to take decision during the transmission of packet from node k to b:

$$f(W_k^+(t), b) = \begin{cases} b \, if \, b \subset W_k(t), \; else \\ i \in W_k^+(t) : d(mi, mb) = min_{j \in W_k^+(t)} d(mj, mb) \end{cases} \tag{3.7}$$

The distance function d is very important for the routing decision process. The Euclidean distance is a best choice. Some other distance functions are described in [26].

3.5.5 *Spray and Focus*

This routing protocol [27] uses the time since two nodes last met each other as context information to calculate the delivery probability of messages. The spraying approaches available in the literature [19, 28], create and spread ("spray") some number of copies of packets to a number of nodes that are within the communication range. Then, each intermediate node stores and carries the packet until it meets the destination or time to live for the packet does not expire. These schemes exploit multiple hops simultaneously and independently for searching the destination; therefore, they are more capable to explore maximum possible routes to destination and can find out the shortest path to the destination.

To gain high performance from such schemes, network's nodes should be highly moveable. However, in many real-life examples, nodes are not highly moveable they are restricted to a small area for the maximum time, e.g., nodes' movement in a university campus. This scenario is more clearly explained in "Spray and Wait" scheme [19, 28]. The working of this scheme is divided into two phases: in the first phase, it transmits some copies of a packet to all nodes within the communication range of it, and in the second phase these neighbor nodes directly transmit the copies of the packet to destination when they encounter. In this way, in this scheme an intermediate node waits for the transmission until it comes within the communication range of destination.

This problem has been solved in Spray and Focus. In the second phase ("focus" phase), node does not wait for direct communication with destination, it gives the packet to more relevant forwarder that has more probability of successful transmission. Now we describe the protocol in detail.

3.5.5.1 Spraying Phase

When a source wants to transmit a message to a destination, Spray and Focus starts "Spray phase" to transmit this message:

Spray Phase: When a source generates a new message for transmission it also creates some copies of the message. These copies of the message are called forwarding tokens. Only the owner of the message can generate and forward forwarding tokens of the message, according to the following rules:

- Each node has a "summary vector" which contains the compact representation of messages currently stored in node's buffer. When two nodes meet each other they exchange summary vector and ask for the messages which are currently absent in it.
- When a node that has a copy of a message and n forwarding tokens of this message meets with another node that does not have a copy of the same message, it creates and gives a copy of that message to the second node and also hand over n/2 forwarding tokens to the second node.
- The message is forwarded according to the "Focus phase", when a node has a single copy of message with only one forwarding token for this message.

3.5.5.2 Focus Phase

When an intermediate node has a message and a single forwarding token of that message, it transmits the message according to "Focus phase". In this phase, message is transmitted to different intermediate nodes according to forwarding policies. These forwarding policies are constructed based on last encounter timers that record time lag from the last meeting of two nodes.

Age of last encounter timers with transitivity: In this protocol, each node i has a timer $T_i(j)$ to record the time elapsed after the last meeting with every other node j in the network. The protocol initially sets $T_i(i) = 0$ and $T_i(j) = \infty$. Whenever i and j meet, both timers are initialized to zero, i.e., set $T_i(j) = T_j(i) = 0$, and increment each timer by 1 at every clock tick. On the basis of these timers, we can construct a utility function, to identify the usefulness of a node to transmit a message to another. Intuitively, messages are transmitted to the nodes that have smaller timer values for the destination.

Single-copy Utility-based Routing: Let each node i has a utility value $U_i(j)$ for every other node j in the network to make the routing more efficient. If a node A wants to transmit a message to node D via node B, then node A can give message to node B only if $U_B(D) > U_A(D) + U_{th}$, where U_{th} is a utility threshold parameter of the algorithm.

The timers become poor identifier as well as their values increase; therefore to improve the efficiency of routing it is essential to minimize the uncertainty for greater timer values. The protocol removes this problem by introducing "transitivity" to update the utility function. When node A meets node B frequently and node B meets node C frequently, due to transitivity rule A can transmit a message to C through B with a high probability, even if A rarely meets with C. Therefore, when node A meets with node B, it also updates its utility values for all nodes for which B has greater utility values. The protocol uses following transitivity function..

Timer Transitivity: Let a node A meets another node B at distance d_{AB}. And $t_m(d)$ represents the time a node takes to travel a distance d while following a given mobility model. Then: $\forall j \neq B : T_B(j) < T_A(j) - t_m(d_{AB}), \text{set} T_A(j) = T_B(j) + t_m(d_{AB})$.

For example the transitivity function for Random Waypoint and Random Walk mobility models are:

$$T_A(D) < T_B(D) - d_{AB}, \text{set} T_B(D) = T_A(D) + d_{AB} (\text{waypoint}), \tag{3.8}$$

$$T_A(D) < T_B(D) - d_{AB}^2, \text{set} T_B(D) = T_A(D) + d_{AB}^2 (\text{walk}), \tag{3.9}$$

These transitivity functions are capable to fast dissemination of utility information in the network and minimize the uncertainty related to location of a particular node.

3.5.6 *BUBBLE Rap Routing*

BUBBLE routing protocol [29] introduces the concept of community in opportunistic network to improve the efficiency of forwarding task. The protocol exploits the community information of nodes to take the forwarding decisions. The protocol follows two major perceptions. First, nodes have varying popularities in community, so the routing protocol prefers the forwarding of message through the nodes that are more popular than current node. Second, there are more chances of interaction between nodes of the same community, so the routing protocol always appreciates the selection of a node as forwarder that belongs to destination community.

The following are two main assumptions of the routing protocol.

- Each node must be related to at least one community. A community can also have only a single node. A node can also belong to several distinct communities.
- Each node has two types of ranking: global ranking and local ranking. Global ranking represents the popularity of the node across the whole communities of the system, while local ranking indicates only popularity of the node within the community to which the node belongs. If a node belongs to several communities, it has several local rankings.

The forwarding task is performed as follows. When a source wants to transmit a message to destination, to forward the message, it constructs a hierarchical ranking tree on the basis of global ranking, until it reaches a node of the destination community. Thereafter, the message is forwarded through local ranking tree which is constructed by local ranking rather than global ranking, until it reaches to destination or time to live of the message expires. In this method, it is not compulsory that every node know the ranking of all other nodes in the network, but they must be capable to compare ranking with neighbor nodes and forward the message to more popular node.

In order to minimize the cost, when a message is transmitted to a node that belongs to destination community, the source node of the message can delete this message from its buffer to make the space and to prevent the further propagation. The forwarding procedure of this routing protocol is picked up from real-life example. For example, if you want to transmit a message to another person, first of all you try to forward the message through neighbor person more popular than you, and then neighbor person forwards it to the more popular person belongs to wider community, e.g., a postman. When the postman encounters a person of the destination community, the message is forwarded to that person. The person who receives the message tries to find out more popular persons within the community. In this way message is forwarded within the destination community, until the message reaches the destination, or the time-to-live expires. Figure 3.2 illustrates the BUBBLE routing protocol.

3.5.7 Integrated Routing Protocol

This routing protocol [30] is an extension of PROPHET routing [20]. Like PROPHET routing protocol, it also uses frequency of meeting between nodes as context information to take the forwarding decision. The integrated routing protocol exploits the mobility pattern of the user to bring the message closer to destination. Real users probably move in a predictable or random fashion, means there are more possibilities of visiting a place by a user that he has visited earlier or he can move to new

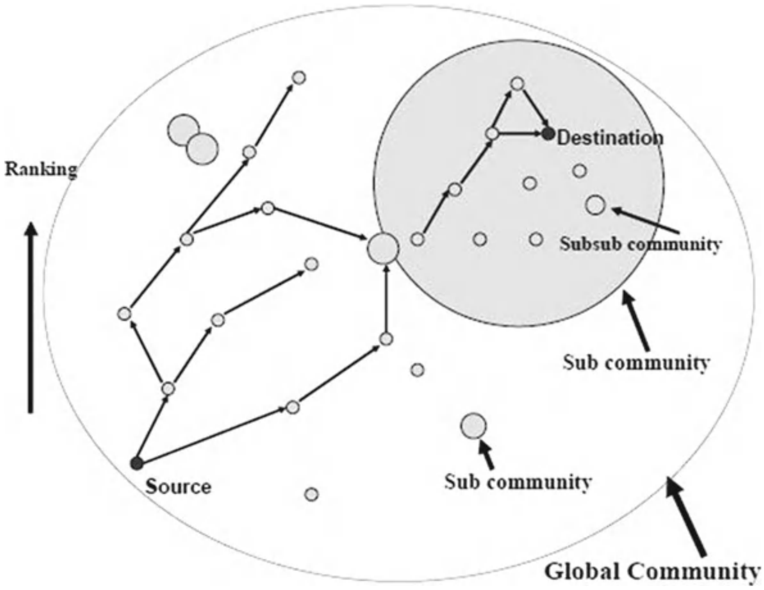


Fig. 3.2 Illustration of the BUBBLE routing protocol

location that he never visited. Therefore, users' mobility behavior may be predictable or unpredictable. The integrated routing protocol exploits this observation to remove the limitations of PROPHET routing protocol. The routing protocol combines the features of flooding-based routing with probabilistic routing and creates a new integrated routing that has the features of both.

To achieve this task, each node maintains the contact probabilities to all other nodes, currently available in the network, in probability matrix as described in [20]. Where, each cell of matrix represents contact probability between two nodes. Each node also maintains a time attribute that represents the time matrix was last updated. Whenever two nodes encounter each other, they update their contact probabilities and time attributes. Nodes exchange their contact probability matrixes and compare them, a matrix is updated by another matrix that has more recent time attribute. Therefore, after the each interaction between two nodes, each node will have same probability matrix.

3.5.7.1 Probability Calculation:

The complete evaluation of delivery probabilities is given in [20]. We update the probability values when two nodes meet, so that the nodes frequently encounter have high message delivery probabilities. When node x encounters node y, the delivery probability of node x for y is updated by Eq. (3.10).

$$P'_{xy} = P_{xy}(1 - P_{xy})P_0 \tag{3.10}$$

where, P_0 represents initial probability. When two nodes do not meet for a long time, the delivery probability between them must be decreased over the time. For example, if nodes x and y do not meet with in a predefined time, the delivery probability is decreased by Eq. (3.11).

$$P'_{xy} = \alpha^k P_{xy} \tag{3.11}$$

where, α is the aging factor, and k represents number of the time units after the last update.

The routing protocol also follows the transitivity rule to compute the delivery probability. According to this, a node x can calculate the delivery probability of node z through node y.

$$P'_{xz} = P_{xz} + (1 - P_{xz})P_{xz}P_{yz}\beta \tag{3.12}$$

where, β is a parameter represents the impact of transitivity.

3.5.7.2 Routing Strategies

When a node gets a message, it checks for a path to destination, path may be available or not. If path is not available, it stores the message in buffer and upon each meeting with other nodes the decisions have to be taken whether or not to forward the message, otherwise the message is further forwarded. Furthermore, a message should be forwarded to the multiple nodes to increase the chances of eventually message delivery to destination. Whenever two nodes meet they exchange their probability matrix. If the receiver is the destination of the message, the message is eventually delivered. Otherwise, if the receiving node has delivery probability greater than a predefine threshold, the message is transmitted to it, if the probability is less than the threshold the message is discarded by the receiver. If all nodes, within the communication range of the sender, do not have context information of destination and the sender has waited for predefined time, the sender disseminates the message to all its' current neighbors like dissemination-based scheme. The same procedure will be continued at each node until the message reaches to destination.

3.6 Fully Context-Aware Routing

Fully context-aware routing protocols provide both facilities, they optimize routing as well as also provide techniques to manage and use context information in efficient way. The routing protocols of this category are more general than the protocols discussed in Sect. 3.5. These protocols can work in every environment, because they can use any set of context information to take routing decisions. The main routing protocols of this category are Context-aware routing [31] and HiBOp [32].

3.6.1 Context-Aware Routing

Communication between nodes in MANET is only possible when nodes are simultaneously connected with same network. In reality, this assumption is mostly not true, because MANET consists of mobile devices that are highly movable. When MANET is highly disconnected it creates a form of opportunistic network. The context-aware routing [31] is a novel protocol to provide communication between such partially connected MANET. Context-aware routing protocol provides communication between different disconnected MANET clouds and assumes an underlying existing MANET routing protocol to connect the nodes within the same cloud. To deliver a message outside the cloud, the sender selects a node from its current cloud that has highest probability to successfully deliver the message to destination. This node buffer the message and waiting either to encounter the destination or any node of destination cloud that has higher delivery probability of encountering the destination. Therefore, nodes calculate delivery probability proactively and spread them in their cloud. But nodes can calculate the delivery probability only for those destinations each node is aware of. The protocol is able to exploit different types of context information, i.e., residual battery life, the probability of meeting between nodes, and the rate of connectivity change. Because the protocol is proactive in nature, each node periodically spreads the information about underlying synchronous routing and a list of delivery probabilities for the other nodes. Each node updates its routing table, when it receives this information.

3.6.1.1 Prediction and Evaluation of Context

The procedure of prediction and evaluation of context information is given below.

- Each node computes its delivery probability to all other nodes of the cloud. And periodically spread them in the network to update the routing information of other nodes.
- Each node maintains a routing table that contains information about next hop and its delivery probability for all known destinations.
- Each node uses local assumption of delivery probability during temporary disconnection and between updates of information.
- If a node does not have any information about the destination of message, it forwards the message to the node that has the highest mobility.
- If an intermediate node encounters a node with higher delivery probability, it forwards the message to higher probability node.

3.6.1.2 Local Evaluation of Context Information

There are various schemes to utilize the multiple types of context information to identify the best forwarder for a particular message. The simplest method is assigned a static value to each type of context information on the basis of priority to create a static hierarchy. However, the assignment of static priorities is inefficient. To make the routing more realistic, the protocol simultaneously maximizes the utilization of several types of context attributes.

Significance-based evaluation of context-aware information: The context information of a node can be represented by a set of attributes (X1, X2, ..., Xn). The attributes represented by a capital letter, e.g. Xn, describe the set of all values a attribute can have, while small letter representation, e.g. x1, is related to specific attribute value from this set. In the protocol, each node is capable to locally create a utility function U(x1, x2, ..., xn) , that represents the delivery probability for every other nodes. The goal of the protocol is, how to select a best suitable node for message delivery, means the node that maximizes all attributes. We can describe the combined goal function as:

$$Maximize\{f(U(x_i)) = \sum_{i=1}^{n} w_i U_i(x_i)\} \tag{3.13}$$

where $w1,\ w2, \ldots, wn$ represent significance weights that describe the importance of each goal.

Autonomic adaptation of the utility function: The protocol is able to accept the weights of each attributes dynamically as soon as the values of these attributes are changed. To do this, the protocol uses a runtime self-adaptation mechanism to adopt the weightings being use for this evaluation process. The protocol modifies the previous formula by adding adaptive weights ai, to make changes in utility function corresponding to disparity of the context.

$$Maximize\{f(U(x_i)) = \sum_{i=1}^{n} a_i(x_i) w_i U_i(x_i)\} \tag{3.14}$$

where, $a_i(x_i)$ may be a composite parameter itself. The following are three important aspects that are very helpful to evaluate its value.

- Difficulty of specific ranges of values, $\mathrm{a}_{\mathrm{range}_\mathrm{i}}(\mathrm{x}_\mathrm{i})$
- Predictability of the context information, $\mathrm{a}_{\mathrm{predictability}_\mathrm{i}}(\mathrm{x}_\mathrm{i})$
- Availability of the context information, $\mathrm{a}_{\mathrm{availability}_\mathrm{i}}(\mathrm{x}_\mathrm{i})$

We now calculate the a_i weights with the help of these factors.

$$a_i(x_i) = a_{range_i}(x_i).a_{predictability_i}(x_i).a_{availability_i}(x_i) \tag{3.15}$$

The protocol models the adaptive weights corresponding to the possible ranges of values $\mathrm{a}_{\mathrm{range}_\mathrm{i}}(\mathrm{x}_\mathrm{i})$ as a function that can contain only [0, 1] . The protocol predicts the behavior of context information by exploiting several statistical attributes, e.g., autocorrelation function to identify the degree of relation between values. Each context attributes has a different degree of availability, whose value may be predictable. Sometimes the values of some attributes are not predictable due to unavailability of sufficient information. The protocol solves this problem by assigning 0 to an adaptive weight ai of missing context information.

$$a_{availability_i}(x_i) = \begin{cases} 1 \text{ if the context information is currently available} \\ 0 \text{ if the context information is not currently available} \end{cases}$$

Automatic adaptation of the refresh period of routing tables and context information: The protocol proposes the following function for the refresh time calculation that is based on the rate of context information dissemination by considering that such information is predictable and mostly the prediction is accurate.

$$t(x1,\ x2,\ \ldots, xn) = c \sum\nolimits_{i=1}^{n} |\rho_{k_i}| \tag{3.16}$$

where, c is a constant of proportionality.

3.6.1.3 Prediction of the Context Information Attributes Using Kalman Filters

Context-aware routing uses Kalman filter [33] for two purposes, to predict the context information of a node more realistically and to optimize the use of bandwidth. Updating the routing tables that store delivery probabilities is a very expensive task, while we can easily predict such information and can use to update the routing tables even when fresh context information is unavailable.

We can represent this prediction problem in the form of a state-space model. We can create a time serious of observed values from context information. It is helpful to create a prediction model with the help of inner state that is described by a set of vectors. The main feature of Kalman filter is that it does not store complete past history of the system that makes it very suitable for the mobile devices with limited storage.

3.6.2 HiBOp (History-Based Opportunistic Routing Protocol)

3.6.2.1 Context Creation and Management

The context in which user lives is made up of two components

- Current context
- Legacy of the context evolution over time

The information about the user itself is known as current context of that user which gets stored in Identity table (IT), as shown in table below. Current context also includes the information about current neighbors of that node. That is why we can call current context as a snapshot of local environment of that user. By using this context information, we can see whether any neighboring node is good forwarder

Table 3.1 Identity table example

Personal information	
Name	ABC
Surname	XYZ
E-Mail	abcxyz@iiitm.ac.in
Phone	+91-9826074XXX
NID	PXLBNGER06CC8Y
Residence	
Street	Morena Road
City	Gwalior
Work	
Street	Morena Road
City	Gwalior
Organization	ABV-IIITM
Work hobbies & fun	
Address	Sport centre
City	Gwalior
Association	BH-2
System information	
MAC-Bluetooth	01:23:45:67:89:AB
MAC-802.11	09:00:07:A9:B2:EB
IP-Address	192.168.1.103

or not. For example, we can see whether there is any neighboring node living in the same area where the destination lives.

Taking decision based on instantaneous information only, does not seems to be good decision, because context information does not tells about users past experiences and his behaviors. For example, user can be called as a good forwarder if he/she meets destination at least once every day. To achieve this there is need to maintain History table, whose structure is like,

Other information which is stored in history table helps HiBOp [32] in calculating probability of meeting that node with destination node in future. HiBOp stores history of more number of encountered nodes than PROPHET. All attributes of encountered nodes provide some legacy in the HiBOp history. Information stored in history table gets refreshed periodically.

Context Management Algorithms: First of all what is identity table? Identity table is a set of personal information, behavioral information, and social information. Personal information is like name and surname behavioral information is like job place and hobbies, system information like IP addresses and MAC addresses. But it is up to the user, which information he/she want to expose in node's IT. HiBOp uses any kind of context information that user provides in IT, but condition is that it should be unique across the whole network. So this is the major advantage of HiBOp. In HiBOp, it is assumed that IT uniquely helps in identifying nodes. Node identity (NID) is the hash of the IT, which is used to uniquely name the each node in the network.

Table 3.2 History table structure

Aggregate	*Class*	*Pc*	*H*	*R*
Gwalior	City	... ..	... ..	... ..

Table 3.3 Repository table structure

Aggr	*Class*	*Carriers*	*ContCount*	*HetCount*	*RedCount*
Gwalior	City	A, B, C	2	1	2

During neighbor discovery phases nodes exchange their IT tables to learn the environment around them. This activity gets carried out periodically and asynchronously. The time interval between two neighbor discovery phases is called signaling interval. At the end of every signaling interval, each node must send its IT or its NID. After signaling interval, if it receives IT or NID of those nodes which were previously present in current context then it marks only its presence and if any new node gets introduced then it broadcast its IT completely. In this way, it exchanges the complete IT's among those nodes, which came in contacts of each other during last signaling interval, whereas stable contacts are just refreshed using NID. During last signaling interval if any node is no more present then its entry from CC table gets removed. To tolerate transitory disconnections, IT is removed from CC table but after death interval (a given no of consecutive signaling intervals).

The second building block is history table, which stores the values of nodes present in ITs of neighbors to whom node met in past. Three counters are used for each aggregate. They are PC, R, and H. Continuity probability (PC) is the probability of encountering the node which carries that value with it, heterogeneity (H) contains average number of nodes which have encountered. H is the fault tolerance index, because high heterogeneity (H) means several distinct chances of encountering aggregate on distinct nodes. Redundancy R is the average number of occurrences of aggregates within the same IT. Redundancy is valuable, because it represents the link toward the aggregate is very high or very low.

As an evolution to the current context, history table gets formed. To update history table dynamically repository table gets used. Repository table stores each node which it sees as its neighbor. After signaling interval repository table gets updated, whereas after flushing interval history table gets updated by merging the data of repository table into history table. After every signaling interval, HiBOp scans current context and for those attributes who do not have row in repository table, new row gets added. Such new attributes get initialized with value as zero. As soon as these attributes found in current context, their counter gets incremented. For each attribute HiBOp follows the following steps:

Increments the continuity counter (continuity counter shows how many times that attribute has been seen as a neighboring node during flushing interval).

If the node whose identity table stores that required attribute is not listed in the carriers list, heterogeneity counter gets incremented and NID gets added to carriers list. If that attribute is there in identity table then its redundancy counter gets incremented.

Flushing interval is an integer number of signaling interval. For each attribute of repository table we calculate continuity probability, heterogeneity and redundancy as follows:

$$P_c^{(rep)} = \frac{ContCount}{M} \tag{3.17}$$

where, M=number of signaling intervals in a repository flushing interval. The continuity probability in the history table is updated as

$$P_c \leftarrow \partial.P_c + (1 - \partial)P_c^{(rep)} \tag{3.18}$$

where, ∂ =classic smoothed average parameter ($0 < \partial < 1$). Similarly, heterogeneity and redundancy are calculated as follows, $H = \partial.H + (1 - \partial)HetCount$

$$R = \partial.R + (1 - \partial)\frac{RedCount}{HetCount} \tag{3.19}$$

$\frac{\text{RedCount}}{\text{HetCount}}$ is a redundancy sample. Dividing RedCount by HetCount is computing the "average redundancy" of the attribute during the flushing interval i.e., the average number of times the attribute has been seen in a single identity table during the flushing interval.

3.6.2.2 Forwarding Operations

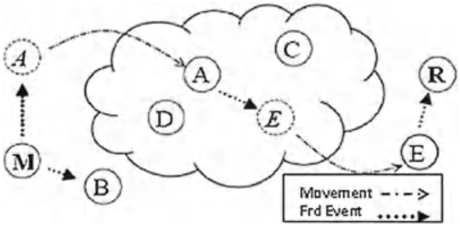
Forwarding is done based on the probability of opportunity to reach destination. Message is passed through the nodes that have higher probability to reach destination. Nothing is new in this scheme. The innovation is on how context can be used to compute delivery probabilities. In this, mainly focus is given on information about destination i.e., message sender includes more information about destination. Sender should include destinations identity table. Match between this information and context information stored at each encountered node is used to calculate delivery probabilities. High probability means high similarity with destination information so ultimately good forwarder.

With this HiBOp controls message replication too, which is a major advantage. Only sender is allowed to replicate the message. Other encountered nodes which carry a message compute delivery probabilities and forward message. When they transfer the message they do not keep copy of forwarded message with them.
The HiBOp works on three phases:

- **Emission**: Sender injects the message in the network and replicates it for reliability.
- **Forwarding**: Exploit's nodes mobility and contacts, each copy of the message proceeds toward destination.
- **Delivery**: When a node finds the destination the process stops.

Delivery phase is of minor importance so not mentioned in detail.

Fig. 3.3 HiBOp forwarding process



Emission Phase: In opportunistic network, it is impractical to maintain reliability using ARQ mechanisms like in MANET. HiBOp tries to maintain reliability using replication at sender only. HiBOp maintains reliability by assuming maximum tolerable message loss which is $\mathrm{P}_1^{\mathrm{max}}$. Using the mechanism written in forwarding phase node calculates delivery probability to reach message up to the final destination. It is denoted by $\mathrm{P}_{\mathrm{succ}}^{(\mathrm{i})}$, where I denotes the ith neighbor ordered by decreasing delivery probability. Delivery probability of sender is $\mathrm{P}_{\mathrm{succ}}^{(0)}$. Assuming these probabilities as an independent, the number of neighbors to which the message is forwarded by the original sender is,

$$k = min\{j \big| \prod\nolimits_{i=0}^{j} (1 - p_{succ}^{i}) \leq p_1^{max}\} \tag{3.20}$$

To keep the joint loss probability below maximum threshold specified by application, sender forwards the message to the minimum number of neighbors. If there are not enough neighbors available, then message is forwarded to available neighbors those are queued at sender. As soon as new neighbors get available, they get used for forwarding. In case of too low delivery probabilities neighbors get used for forwarding if and only if there delivery probability is above 0.001.

Forwarding Phase:

a. Weighing attributes

The main idea of HiBOp is calculating delivery probabilities based on matches between the sender information in the message and the destination's context information available on nodes [32]. Matches should be weighted based. Class weights should represent the precision of that class in identifying the destination. Match on the destination's name gives more precise information than a match on the residence city of the destination.

Several functions can be used to assign weights to classes based on the ranking. For this case named w0, we have computed other weights as $w_{i+1} = w_i + r_i, \beta$, where β is defined as weight increase parameter and ri is the maximum redundancy of the ith class.

The main idea is,

1. Weight should be monotonically increasing.
2. Relative difference between classes should increase if the less significant one allows for a higher redundancy, because higher redundancy usually means lower significance.

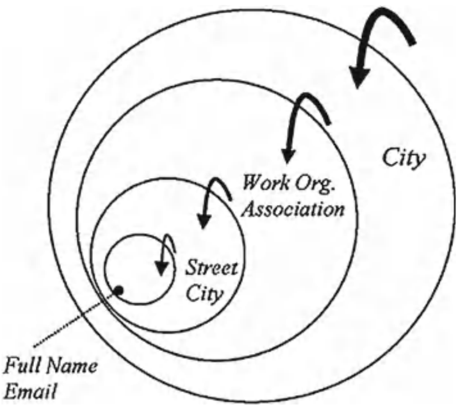


Fig. 3.4 Precision of attribute classes

b. Forwarding based on Delivery Predictability

A node wishing to forward message, broadcasts the context information of destination by attaching its own delivery probability to reach destination. Nodes that receive such message evaluate their delivery probability and send it reverse to the questioning node if it is higher.

At each node, delivery probability gets calculated from three steps:

- Node's Identity Table,
- Node's current context,
- Node's history.

Now, let us see how HiBOp exploits current context to calculate the delivery probability.

Node finds those attributes in its identity table that matches with attributes of destination. From identity table, delivery probability is calculated by taking the ratio between sum of weights of matching attributes and sum of weights of attributes specified in the destination information i.e.,

$$P_{IT} = \frac{\sum_{j\in\{match\}} W_j}{\sum_{j\in\{dst\_info\}} W_j} \tag{3.21}$$

We know that current context is made up of identity tables of current neighbors. For each neighbor we calculate PIT, and the delivery probability related to the current context is the maximum over these probabilities:

$$P_{cc} = \max j \in CCP_{IT}^{(j)} \tag{3.22}$$

Using history to calculate delivery probability requires few more steps. Each aggregate in history table has three indices, continuity probability (PC), heterogeneity (H), and redundancy (R). HiBOp selects those aggregates that match with destination information. For these aggregates R and PC indices are combined as follows:

$$P_{op}^{(j)} = P_c^{(j)}.\frac{R^{(j)}}{r} \tag{3.23}$$

where, r= maximum possible redundancy for the class of j-th matching aggregate. Pc is scaled according to potential redundancy that the aggregate could achieve. Similarly, contribution related to history table is calculated based on the weighted mean of the $P_{op}^{(j)}$ values,

$$P'_H = \frac{\sum_{j\in\{match\}} P_{op}^{(j)} W_j}{\sum_{j\in\{dst\_info\}} W_j} \tag{3.24}$$

The delivery probability of history is calculated by making changes in P'_H according to H indices of matching attributes. HiBOp increases Δ_{max} at most, scaling P'_H according to the average heterogeneity of matching attributes ($\bar{\mathrm{h}}$):

$$P_H = \max\{1,\ P'_H + \Delta_{max}.[1 - e^{-(\bar{h}-1)}]\} \tag{3.25}$$

Now, note since Δ_{max} is scaled according to an exponential law, same average heterogeneity increase results in a higher PH increase for small values of heterogeneity.

The delivery probability is finally computed by combining equations of PIT, PCC, and PH,

$$P = \alpha.P_H + (1-\alpha).\max\{\eta.P_{cc}, P_{IT}\} \tag{3.26}$$

This equation is made up of two components, weighted with smoothing factor $\alpha(0 \le \alpha \le 1)$. The components are,

- PH which describes the legacy of the node's history.
- Second component describes the current status of the node's environment.

The α gives more weight to past history or to the current environment. The node's current environment is described by PIT and PCC, so they are combined in last equation. The η factor (η $<$ 1) scales down PCC with respect to PIT, because PCC is related to neighbor, while PIT is related to the local node.

3.7 Conclusions

In this chapter, we have represented the meaning and utilization of context information to make the routing more efficient in opportunistic network. We have described how various approaches of routing in opportunistic network exploit this context information to automatically adopt the dynamic environment resulting from mobility of mobile nodes. We have also presented here a classification of the routing protocols of opportunistic network on the basis of amount of context information they exploit to take the routing decisions. And we explained the main routing protocols of both partially context-aware and fully context-aware routing categories. The opportunistic network is automatically constructed by users' mobile devices, developing a social relationship model to map the users' social relationship and using them for constructing routing protocols is a very emerging research domain. Another interesting research direction is how to merge completely infrastructure-less opportunistic network and infrastructure-full network. Finally, designing some concepts to provide the facilities of real-time communication applications in opportunistic networks is another extremely important direction.

References

1. Borgia E, Conti M, Delmastro F, Pelusi L (2005) Lessons from an Ad-Hoc network test-bed: middleware and routing issues. Ad Hoc Sens Wirel Netw 1:7
2. Pelusi L, Passarella A, Conti M (2006) Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. IEEE Commun Mag 44:134–141
3. Conti M, Kumar M (2010) Opportunities in opportunistic computing. J Comput 43(1):42–50
4. Fall K (2003) A delay-tolerant network architecture for challenged internets. Paper presented in the proceedings of the 2003 ACM conference on applications, technologies, architectures, and protocols for computer communications (SIGCOMM), pp 27–34
5. Jain S, Fall K, Patra R (2004) Routing in a delay tolerant network. Paper presented in the proceedings of the 2004 ACM conference on applications, technologies, architectures, and protocols for computer communications (SIGCOMM), ACM, New York, pp 145–158
6. Boldrini Chiara, Conti Marco, Passarella Andrea (2008) Autonomic behaviour of opportunistic network routing. Int J Auton Adapt Commun Syst 1(1):122–147
7. Conti M, Giordano S (2007) Multihop Ad Hoc networking: the reality. IEEE Commun Mag 45:88–95
8. Conti M, Giordano S (2007) Multihop Ad Hoc networking: the theory. IEEE Commun Mag 45:78–86
9. Brown P, Bovey J, Chen X (1997) Context-aware applications: from the laboratory to the marketplace. IEEE Pers Commun 4(5):58–64
10. Ryan N, Pascoe J, Morse D (1997) Enhanced reality fieldwork: the context-aware archaeological assistant, In: Gaffney V et al. (eds) Computer applications in archaeology
11. Schmidt A, Aidoo K, Takaluoma A, Tuomela U, Van Laerhoven K, Van de Velde W (1999) Advanced interaction in context, 1th international symposium on handheld and ubiquitous computing (HUC99), Lecture notes in computer science, vol 1707. Springer, pp 89–101
12. Dey A, Abowd G (2000) Towards a better understanding of context and context-awareness. Proceedings of the workshop on the what, who, where, when and how of context-awareness, affiliated with the CHI 2000 conference on human factors in computer systems, New York
13. Dey A, Salber D, Abowd G (2001) A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. Hum Comput Interact (HCI) J (special issue on context-aware computing) 16(2–4):97–166
14. Conti M, Crowcroft J, Giordano S, Hui P, Nguyen HA, Passarella A (2008) Minema. Miranda H, Rodrigues L, Garbinato B (eds) Routing issues in opportunistic networks. Springer, Berlin
15. Jindal A, Psounis K (2007) Contention-aware analysis of routing schemes for mobile opportunistic networks. Paper presented in the proceedings of the 1st international ACM MobiSys workshop on mobile opportunistic networking (MobiOpp 2007), pp 1–8
16. Vahdat A, Becker D (2000) Epidemic routing for partially connected Ad Hoc networks. Technical report CS-2000-06, Duke University, Computer Science Department

17. Widmer J, Le Boudec J (2005) Network coding for efficient communication in extreme networks. Paper presented in the proceedings of the ACM SIGCOMM 2005 workshop on delay tolerant networking (WDTN 2005), pp 284–291
18. Chen L-J, Yu C-H, Sun T (2006) A hybrid routing approach for opportunistic networks, SIGCOMM'06 workshops. Pisa, Italy, pp 213–220
19. Spyropoulos T, Psounis K, Raghavendra CS (2005) Spray and wait: efficient routing in intermittently connected mobile networks. In proceedings of ACMSIGCOMM workshop on delay tolerant networking (WDTN)
20. Lindgren A, Doria A, Schelen O (2003) Probabilistic routing in intermittently connected networks. ACM Mob Comput Commun Rev 7:19–20
21. Burns B, Brock O, Levine B (2005) MV routing and capacity building in disruption tolerant networks. Paper presented in the proceedings of the 24th IEEE annual joint conference of the IEEE computer and communications societies (INFOCOM 2005)
22. Davis J, Fags A, Levine B (2001) Wearable computers and packet transport mechanism in highly partitioned ad hoc networks. In Proceedings of Inti. Synylusirrni on Wrcrrabk Cnnrpvters
23. Burgess J, Gallagher B, Jensen D, Levine B (2006) MaxProp: routing for vehicle-based disruption-tolerant networks. Paper presented in the proceedings of the 25th IEEE annual joint conference of the IEEE computer and communications societies (INFOCOM 2006)
24. Jain S, Fall K, Patra R (2004) Routing in a delay tolerant network. In Proc. ACM SIGCOMM, pp 145–158
25. Hui P, Chaintreau A, Scott J, Gass R, Crowcroft J, Diot C (2005) Pocket switched networks and human mobility in conference environments, in Proc. WDTN, 2005
26. Leguay J, Friedman T, Conan V (2005) DTN routing in a mobility pattern space, in Proc. WDTN
27. Spyropoulos T, Psounis K, Ragavendra C (2007) Efficient routing in intermittently connected mobile networks: the multiple-copy case. ACM/IEEE Trans Netw 16:77–90
28. T. Small and Z. Haas, Resource and performance tradeoffs in delay-tolerant wireless networks, In Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005
29. Hui P, Crowcroft J (2007) Bubble rap: forwarding in small world dtns in every decreasing circles. Technical report, Technical report UCAM-CL-TR684. University of Cambridge, Cambridge, UK
30. Verma Anshul (2011) Dr. Anurag Srivastava, Integrated routing protocol for opportunistic networks, (IJACSA). Int J Adv Comput Sci Appl 2(3):85–92
31. Musolesi M, Hailes S, Mascolo C (2005) Adaptive routing for intermittently connected mobile ad hoc networks. Paper presented in the proceedings of the IEEE international symposium on a world of wireless, mobile and multimedia networks (WoWMoM 2005), pp 183–189
32. Boldrini C, Conti M, Passarella A (2007) Impact of social mobility on routing protocols for opportunistic networks. Paper presented in the proceedings of the first IEEE WoWMoM workshop on Autonomic and Oppoortunistic Networking (AOC 2007)
33. Kalman RE (1960) A new approach to linear filtering and prediction problems. Transactions of the ASME J Basic Eng 82:34–45

Chapter 4
Energy-Latency Tradeoff of Opportunistic Routing

Ruifeng Zhang, Olivier Berder and Olivier Sentieys

Abstract Opportunistic routing aims at exploiting sporadic radio links to improve the connectivity of multi-hop networks and to foster data transmissions. However, the benefit of opportunistic relaying may be counteracted due to energy increase resulted from multiple active receivers. In this chapter, we propose a thorough analysis of opportunistic relaying efficiency under the different realistic radio channel conditions. The study is intended to find the optimal tradeoff between two objectives: energy and latency minimizations, with a hard reliability constraint. We derive the optimal bound, namely, the Pareto front of the related optimization problem, which offers a good insight into the advantages of opportunistic routing compared with classical multi-hop routing schemes. Moreover, the experiment and simulation results verify this optimal bound. The study of lower bound provides a framework to optimize the parameters at the physical, MAC, and routing layers during the design or planning phase of a network from a cross-layer viewpoint

Keywords Energy-latency tradeoff · Pareto front · Opportunistic routing · Cross-layer.

4.1 Introduction

Opportunistic networking refers to all techniques that benefit from the use of spontaneous radio links in mobile ad hoc networks (MANETs) [1]. In the traditional networking, the link between pairwise nodes is distinguished as being connected or

---

R. Zhang (✉) · O. Berder · O. Sentieys
IRISA, INRIA, Université de Rennes 1, Rennes, France
e-mail: ruifeng.zhang@inria.fr

O. Berder
e-mail: olivier.berder@irisa.fr

Sentieys
e-mail: olivier.sentieys@irisa.fr

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_4,

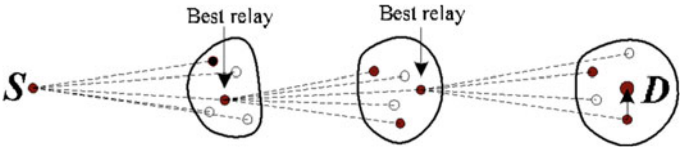


Fig. 4.1 Principle of opportunistic routing

not. Actually, this kind of classification relies on the traditional layered design of networking: the physical and MAC layers ensure a perfect connection of some preselected links and the routing layer selects the path from these links. However, such unreliable links can be rather beneficial in the opportunistic networking if strong channel states are opportunistically exploited.

The opportunistic forwarding techniques [2, 3] employ the same principle, i.e., spatial diversity, to improve the multi-hop transmissions. The main idea of opportunistic forwarding is that at each hop, a set of next-hop relay candidates that have successfully received a packet compete for acting as relay. In the phase of relay-selection, a priority is assigned to each relay candidate according to a predefined metric, for example, the geographical closeness of the relay candidate to the destination [4], as shown in Fig. 4.1.

The objective of this chapter is to evaluate the maximal efficiency of such opportunistic routing schemes that can be achieved, i.e., to find an optimal set of forwarding nodes to balance the energy cost and transmission delay. The efficiency of opportunistic routing can be evaluated from different viewpoints. With respect to the multi-hop networks, we identify three important performance parameters: the end-to-end reliability, the end-to-end delay (referred to as latency hereafter), as well as the energy consumption. Accordingly, to evaluate the opportunistic routing completely, we introduce a multi-objective optimization framework [5]. Due to the fundamentality of reliability, we consider it as a hard constraint in this chapter. The use of acknowledged transmissions allows to fulfill this constraint, at least from a theoretical point of view. The other two constraints, i.e., energy and latency, are considered as two competing objective functions that should be simultaneously minimized.

Some analytical models have been proposed for opportunistic routing in [6, 7]. The modeling framework in [6], which separates the opportunistic routing functionality into three components: routing, medium access, and sleep discipline, is presented to analyze the energy efficiency and latency performance of opportunistic routing in low traffic scenarios. This framework rests on the disc model [8] that is based on the definition of a radio power reception threshold and cannot take the realistic channel features into account, such as fading or shadowing. In [7], the effect of the shadowing and fading is taken into account in the proposed model, and the expected effective transmission distance (ETD) is defined. However, this model still relies on the definition of a reception threshold, represented by the signal-to-noise (SNR) threshold in the switched link. Both the disc and the switched link models are not realistic and have severe weakness that is particularly relevant to opportunistic routing.

In fact, due to the complexity of wireless environment and possible movement of nodes or surrounding objects, a network is an evolving process with unstable links and always remaining in an intermediate state, that is, partially connected, as described in [9–12]. Moreover, these works show that the unreliable links can be effectively exploited to improve the performance of the MAC or routing layer. More specifically, opportunism may improve the performance of multi-hop routing by selecting one relay according to its instantaneous linking status [8].

In the classical opportunistic routing, the neighbors of a source within a certain area act as relay candidates. Several existing works on the classical opportunistic routing, such as [2, 6, 13, 14], provide the analysis of energy and latency performance. In [2, 6], the energy and latency performance of a routing scheme called *GeRaF* are analyzed. Meanwhile, the effects of node density, traffic load, and duty cycle are evaluated. The simulation results in [13] demonstrate the impacts of node density, radio channel quality, and traffic rate on the energy consumption per node, the average delay of a packet, and the goodput of the opportunistic protocol. It is concluded that the benefit of opportunistic scheme is about 10 % decrease in energy expenditure and 40 % reduction in delay. Besides, the energy efficiency of the protocol CAGIF is studied in the scenario of fading channel in [14]. Nevertheless, these analyses are based on an unrealistic disc link model or a switch link model. Furthermore, the schemes in which the whole set of neighbor nodes try to receive packets from the source node degrade the energy performance, as proved in [15].

With the aim of improving the energy efficiency, an efficient selection mechanism of relay nodes is proposed in [4], rather than choosing all the neighbors as relay candidates. The corresponding simulation results in a shadowing channel indicate that the energy efficiency is greatly enhanced. However, in the aforementioned studies, a fixed transmission power is considered, and the number of relay candidates is chosen according to a given routing policy without providing any proof of optimality. Therefore, these studies are insufficient to determine whether the relative low performance of opportunistic routing is intrinsic to this kind of routing or due to the specific protocol (relay selection policy, fixed power choice, etc.).

In this context, we propose in this chapter to analyze the Pareto front of the energy-latency tradeoff for opportunistic routing under a hard end-to-end reliability constraint. To obtain this bound, we consider the size of candidate cluster and the transmission power as variables of the optimization problem. Instead of designing a specific relay selection mechanism, we focus on the following two issues: what is the best set of relay candidates and what is the performance achieved by the optimized set of candidates?

With respect to the routing policy, we assume that for a given cluster, only the candidate closest to the destination is selected to forward the packet. Such a strategy obviously relies on the assumption that each node has the full knowledge of the position of itself and the destination. Once a node has a packet to send, it appends the location information of itself and the intended relay cluster to the packet and then broadcasts it. The relay candidates that successfully receive the packet (described as solid circles in Fig. 4.1) assess their own priorities of acting as relay, based on how close they are to the destination. The *best relay* that is the closest to the destination forward the packet illustrated in Fig. 4.1. Compared with the aforementioned schemes, this scheme utilizes an optimized candidate cluster, rather than all active neighbors, to receive the packet for the purpose of saving energy and taking advantage of the spatial diversity.

The main contributions of this chapter are summarized as follows:

- A general framework for evaluating the maximal efficiency of the opportunistic routing principle is provided. Energy and latency are compromised under the constraint of end-to-end reliability.
- The Pareto front of energy-latency tradeoff is derived in different scenarios. A closed-form expression of energy-latency tradeoff is obtained when the number of relay candidates is fixed. Moreover, an algorithm to find the optimal number of relay candidates is proposed. The simulation results verify the correctness of this lower bound in a two-dimension Poisson distributed network. The numerical analysis shows that the opportunistic routing is inefficient in the Additive White Gaussian Noise (AWGN) channel; however, it is efficient in the Rayleigh block fading, and Rayleigh fast fading channels under the condition of a small-size cluster.
- The lower bound of energy efficiency and its corresponding maximal delay are derived.

The remaining part of this chapter is organized as follows: Section 4.2 describes the models and metrics used in this chapter. In Sect. 4.3, the lower bound of energy-delay tradeoff of one-hop transmission is deduced and the lower bound of energy efficiency is obtained for the delay-tolerant applications. The results about one-hop transmission are extended to the scenario of multi-hop transmissions in Sect. 4.4. Furthermore, the related gain of opportunistic routing on the energy efficiency is analyzed. In Sect. 4.5, a novel opportunistic protocol is introduced and the theoretical analysis is verified by the simulation results. Section 4.6 discusses the significance of these results and gives some conclusions.

4.2 Models and Metrics

This section introduces the problem formulation by modeling wireless links, energy consumption, and transmission delay. Two metrics are proposed and referred to as mean energy distance ration per bit and mean delay distance ratio. For the readability, all parameters used in this chapter are described in Table 4.1. The related values of parameter configurations are obtained from the datasheet of [16].

Table 4.1 Notations

Symbol	Description	Value	Symbol	Description	Value
α	Pathloss exponent	3	N_{head}	Number of bits in the overhead of a data packet	0
β_{amp}	Amplifier proportional offset (>1)	14	N_R	Number of cooperative receivers	
γ	Signal-to-Noise ratio		N_{Ropt}	Optimal number of cooperative receivers	
τ_{ack}	ACK ratio	0.08125	N_{R0}	Minimum number of cooperative receivers	
τ_{head}	Overhead ratio		pl	Link probability	
b	Number of bits per symbol	$\log_2 M$	pl_{opt}	Optimal link probability	
B	Bandwidth of channel	250 KHz	P_0	Minimum optimal transmission power	
d_0	Minimum transmission distance		P_{opt}	Optimal transmission power	
d_{hop}	One-hop transmission distance		P_{rxElec}	Receiver circuitry power	59.1 mW
f_c	Carrier frequency	2.4 GHz	P_{start}	Startup power	38.7 mW
G_{Rant}	Receiver antenna gain	1	P_t	Transmission power	
G_{Tant}	Transmitter antenna gain	1	P_{txElec}	Transmitter circuitry power	
L	Circuitry loss	1	R_b	Bit rate	250 Kbps
M	Modulation order		R_{code}	Code rate	1
N_0	Noise level	-150 dBm/Hz	R_s	Symbol rate	
N_{ack}	Number of bits in an ACK packet	78	T_{queue}	Delay from queuing	
N_b	Number of bits per packet	2560	T_{start}	Startup time	0

4.2.1 System Model

In this chapter, the nodes in a network are assumed to be independently and randomly distributed according to a random Poisson process of density ρ. The probability of n nodes locating in a region $\mathcal{A}$ follows a two-dimension Poisson distribution:

$$P(N \text{ nodes in } \mathcal{A}) = \frac{(\rho \cdot S_A)^N}{N!} \exp(-\rho \cdot S_A), \tag{4.1}$$

with $E(N) = \rho \cdot S_A$. Here, S_A is the surface of $\mathcal{A}$ and $\exp(\cdot)$ represents the exponential function.

We consider the case when a source node $\mathcal{S}$ forwards a packet to a sink/destination node $\mathcal{D}$. We assume that n_i is one of $\mathcal{S}$'s neighbors which is closer to $\mathcal{D}$ than $\mathcal{S}$. In addition, each n_i is associated with a pair (pl_i, d_i), where pl_i is the link probability between n_i and $\mathcal{S}$ described in (4.10), and d_i is the effective transmission distance given by

$$d_i = d_{\mathcal{S}\mathcal{D}} - d_{n_i\mathcal{D}}. \tag{4.2}$$

Here, $d_{\mathcal{S}\mathcal{D}}$ and $d_{n_i\mathcal{D}}$ are the Euclidian distance between $\mathcal{S}$ and $\mathcal{D}$ and that between n_i and $\mathcal{D}$ respectively.

4.2.2 Energy Consumption Models

We denote the forwarding candidate set as $\mathcal{F}$ which includes all the nodes involved in the local collaborative forwarding. The number of nodes in $\mathcal{F}$ is N_R. According to the assumptions of routing policy in Sect. 4.1, the energy consumption for transmitting one packet E_p is composed of three parts: the energy consumed by the transmitter E_{Tx}, by the receiver E_{Rx}, and by the acknowledgment packet exchange E_{ACK}, i.e.,

$$E_p = E_{Tx} + N_R \cdot E_{Rx} + E_{ACK}. \tag{4.3}$$

In this work, these energy consumption components are expressed by utilizing the models proposed in [17] as follows:

$$E_{Tx} = T_{start} \cdot P_{start} + \frac{N_b + N_{head}}{R_b \cdot R_{code}} .(P_{txElec} + \beta_{amp} \cdot p_t), \tag{4.4}$$

$$E_{RX} = T_{start} \cdot P_{start} + \frac{N_b + N_{head}}{R_b \cdot R_{code}} \cdot P_{rxElec}. \tag{4.5}$$

We further assume that the same transmission scheme is used for data and acknowledgment transmissions, which allows another assumption that the acknowledgment energy is a fraction of the data transmission one

$$E_{ACK} = \tau_{ack}(N_R \cdot E_{RX} + E_{TX}), \tag{4.6}$$

where $\tau_{ack} = (N_{ack} + N_{head})/(N_b + N_{head})$ represents the length's ratio between ACK and data packets. We also assume that an ACK packet is smaller than a data packet [17], i.e., $0 \leq \tau_{ack} < 1$. Therefore, the energy consumption per bit defined by $E_b = E_P/N_b$ is given by

$$E_b = E_c + K_1 \cdot P_t, \tag{4.7}$$

where $(K_1 \cdot P_t)$ and E_c represent the radio emission energy and the circuit energy per node, respectively. Both are obtained by substituting (4.4), (4.5), and (4.6) into (4.3) as follows:

$$E_c = (1 + \tau_{ack}) \cdot \left((N_R + 1) \cdot \frac{T_{start} \cdot P_{start}}{N_b} + (1 + \tau_{head}) \cdot \left(N_R \cdot \frac{P_{rxElec}}{R_b \cdot R_{code}} + \frac{P_{rxElec}}{R_b \cdot R_{code}} \right) \right), \tag{4.8}$$

$$K_1 = (1 + \tau_{ack}) \cdot (1 + \tau_{head}) \cdot \frac{\beta_{amp}}{R_b \cdot R_{code}}, \tag{4.9}$$

where $\tau_{head} = N_{head}/N_b$. The related parameters are described in Table 4.1.

4.2.3 Realistic Unreliable Link Models

As claimed in Sect. 4.1, transmission errors play an important role in reliable communications. Hence, we consider herein the radio link probability as the metric of link quality, which is derived from the packet error rate (PER) according to [17]

$$pl(\gamma_{x,x'}) = 1 - PER(\gamma_{x,x'}), \tag{4.10}$$

where $PER(\gamma)$ is the PER obtained according to a signal-to- noise ratio (SNR) $\gamma \cdot \gamma_{x,x'}$ is the SNR between nodes x and x', which is obtained from the classical attenuation model in [17]

$$\gamma_{x,x'} = K_2 \cdot P_t \cdot d_{x,x'}^{-\alpha}, \text{ with } K_2 = \frac{G_{Tant} \cdot G_{Rant} \cdot \lambda^2}{(4\pi)^2 N_0 \cdot B \cdot L}. \tag{4.11}$$

Here, $d_{x,x'}$ denotes the transmission distance between nodes x and x' and B is equal to R_s for the sake of simplicity. Refer to Table 4.1 for other parameters.

PER may have various forms depending on the employed transmission technology (modulation, coding, diversity, etc.). We approximate the unreliable link models in the AWGN, Rayleigh fast fading, and Rayleigh block fading channels, respectively, which are described as follows (refer to [18] for more details):

- AWGN channel

$$pl_g(\bar{\gamma}) = (1 - 0.826\alpha_m \cdot \exp(-0.5415\beta_m\bar{\gamma}))^{N_b}, \text{if } \beta_m \cdot \bar{\gamma} \geq 2; \tag{4.12}$$

- Rayleigh fast fading channel

$$pl_f(\bar{\gamma}) = \left(1 - \frac{\alpha_m}{2\beta_m\bar{\gamma}}\right)^{N_b}; \tag{4.13}$$

- Rayleigh block fading channel

$$pl_b(\bar{\gamma}) = \exp\left(\frac{-4.25\log_{10}N_b + 2.2}{\beta_m\bar{\gamma}}\right), \text{when } \alpha_m = 1. \tag{4.14}$$

Here, α_m andrely β_m on the modulation type and order. For instance $\alpha_m = 1$, and $\beta_m = 2$ in the case of Binary Phase Shift Keying (BPSK), and $\alpha_m = 4(1 - 1/\sqrt{M})/\log_2(M)$, and $\beta_m = 3\log_2(M)/(M-1)$ for Multiple Quadrature Amplitude Modulation (MQAM).

Regarding the opportunistic relaying principle, a successful transmission means that at least one node receives the packet correctly. Therefore, the probability of a successful transmission in the scenario of N_R forwarding nodes is expressed as

$$p_s = \sum_{i=1}^{N_R} pl_i \prod_{j=1}^{i-1}(1 - pl_j); \tag{4.15}$$

where pl_i is the link probability betweenand nodedefined in (4.11). It should be noted that the priority sequence of the N_R forwarding candidates influences the value of p_s and its effect is described in [4].

To ensure reliable transmissions, a retransmission procedure with an acknowledgment mechanism is adopted. The scheme that will retransmit infinitely until reaching a successful transmission is assumed in this work. Accordingly, the average transmissions number $\bar{N}_{TX}$ is the sum of a geometric series represented by

$$\bar{N}_{TX} = \sum_{n=1}^{\infty} n \cdot p_{sdata} \cdot p_{sack} \cdot (1 - p_{sdata} \cdot p_{sack})^{(n-1)} = \frac{1}{p_{sdata} \cdot p_{sack}}, \tag{4.16}$$

where n is the number of transmissions, p_{sdata} and p_{sack} are the successful transmission probability of data packet and ACK packet, respectively, calculated by (4.15).

The ACK transmission failures are assumed to be neglected (i.e.,$p_{sack} = 1$) in this work, which would facilitate our derivation. This assumption is on the grounds of the following three aspects: (i) Both data and ACK packets are assumed to experience the same channel state [19]. (ii) ACK packets are supposed to be smaller than data packets. For instance, in the case when the length of data packet and ACK is 320 and 26 bytes, respectively, the successful transmission probability of data packets achieves 80 %; meanwhile, the corresponding successful probability can attain up to 98 % for ACK in the same channel conditions. (iii) A more efficient coding scheme can be employed for ACK packets to improve their reliability if necessary. Under this assumption, $\bar{N}_{Tx}$ is thus approximated as

$$\bar{N}_{TX} \approx \frac{1}{p_s}, \tag{4.17}$$

where p_s is equal to p_{sdata}.

4.2.4 Mean Energy Distance Ratio Per Bit

Due to the fact that the energy cost monotonically increases with the transmission distance, the metric of mean Energy Distance Ratio per bit ($\overline{EDRb}$) (in J/m/bit), as proposed in [20], is introduced into this work. This metric indicates the energy consumption for transmitting one bit over one meter. According to this definition, we have

$$\overline{EDRb} = \frac{E_b(P_t) \cdot \bar{N}_{TX}}{\bar{d}_{tx}}, \tag{4.18}$$

where $\bar{d}_{tx}$ is the expected transmission distance for an opportunistic transmission. The effective distance depends on the retransmission priority policy. After arranging the priorities of relay candidates in the descending order, the average effective distance is derived by

$$\bar{d}_{TX} = \frac{1}{p_s} \cdot \sum_{i=1}^{N_b} d_i \cdot pl(d_i, P_t) \prod_{j=1}^{i-1} (1 - pl(d_j, P_t)). \tag{4.19}$$

Substituting (4.7), (4.17), and (4.19) into (4.18) yields

$$\overline{EDRb} = \frac{E_c + K_1 P_t}{\sum_{i=1}^{N_R} di \cdot \prod_{j=1}^{i-1} (1 - pl(d_j, P_t))}. \tag{4.20}$$

It should be noted that this metric integrates all physical and link layers parameters. Consequently, it is applicable to the synthetical analysis of the efficiency of physical and MAC layers.

4.2.5 Mean Delay Distance Ratio

The delay of a packet to be transmitted over one hop, D_{hop}, is defined as the sum of three delay components. The first component corresponds to the queuing delay $T_{queuing}$, during which a packet waits for being transmitted. The second component is the transmission delay that is equal to $N_b/(R_{TX}R_{code})$. The third component is T_{ACK}. Note that we neglect the propagation delay because the transmission distance between two nodes is usually short in multi-hop networks. Furthermore, a reliable one-hop transmission suffers from the delay caused by retransmissions. According to (4.17), the mean delay of a reliable one-hop transmission is $\overline{D}_{hop} = D_{hop}\overline{N}_{TX}$.

Since delay rises with the increase of transmission distance, we propose a new delay metric, i.e., the Delay Distance Ratio ($\overline{DDR}$). It is defined by

$$\overline{DDR} = \frac{D_{hop}\overline{N}_{TX}}{\bar{d}_{TX}} = \frac{D_{hop}}{\sum\limits_{i=1}^{N_R} d_i \cdot pl(d_i, P_t) \prod\limits_{j=1}^{i-1} (1 - pl(d_j, P_t))}. \tag{4.21}$$

Note that $\overline{DDR}$ includes all factors of physical and link layers similarly. Therefore, $\overline{EDRb}$ and $\overline{DDR}$ are the effective metrics to measure the impact of these parameters on the energy efficiency and delay of a network.

4.3 Nergy-Delay Tradeoff for One-Hop Transmission

In this section, we analyze the energy-latency tradeoff under the reliability constraint in the scenario of one-hop transmission. The optimal transmission power and the optimal number of receivers will be analyzed and the closed-form expression of lower bound of energy-delay tradeoff is obtained.

The energy-delay tradeoff of one-hop transmission can be abstracted as a constrained optimization problem

$$\text{Minimize:}\overline{EDRb}$$

$$\text{Subject to}: N_R \geq 1, N_R \in \text{Integer}, \overline{DDR} \leq ddr.$$

Here, *ddr* represents the delay constraint. Consequently, minimizing the energy consumption under a delay constraint can be achieved by optimizing the three parameters ($P_{opt}, N_{Ropt}, d_{iopt}(i = 1, \ldots N_R)$) for one-hop transmission. Therein, P_{opt} is the optimal transmission power, N_{Ropt} is the optimal number of opportunistic relay candidates, and d_{iopt} is the optimal transmission distance for each relay candidate.

This is a mixed integer nonlinear programming (MINLP) problem that can be solved using a branch-and-bound algorithm [21], but it is time consuming. Thus, we propose in the following an alternative which relies on an analytic solution when the size of the forwarding set is constant. A simple iterative procedure is then proposed to find the optimal size.

4.3.1 Energy-Delay Tradeoff for a Given Number of Receivers

According to (4.11), the physical transmission distance is related to the SNR and is expressed as $d_{hop} = (k_2 p_t/\gamma)^\alpha$. Since the forwarding nodes may locate in the vicinity of the line between $\mathcal{S}$ and $\mathcal{D}$, rather than exactly over the $\mathcal{S}$-$\mathcal{D}$ line, the physical distance is smaller than or equal to the related effective distance. Neverthess, we assume that the physical and effective distances are equal with the purpose of minimizing $\overline{EDRb}$ which means that the forwarding nodes are precisely located on the $\mathcal{S}$-$\mathcal{D}$ line. Under this approximation, (4.20) and (4.21) are converted to be the functions of P_i and $\bar{\gamma}_i$ as follows:

$$\overline{EDRb}(\bar{\Gamma}, P_t, N_R) = \frac{E_c(N_R) + K_1 P_t}{(K_2 P_t)^{1/\alpha}} \cdot g(\bar{\Gamma}, N_R), \tag{4.22}$$

$$\overline{DDR}(\bar{\Gamma}, P_t, N_R) = \frac{D_{hop}}{(K_2 P_t)^{1/\alpha}} \cdot g(\bar{\Gamma}, N_R), \tag{4.23}$$

where $\Gamma = [\bar{\gamma}_1, \ldots, \bar{\gamma}_i, \ldots \bar{\gamma}_{N_R}]$ and

$$g(\bar{\Gamma}, N_R) = \frac{1}{\sum_{i=1}^{N_R} \bar{\gamma}_i^{-\frac{1}{\alpha}} \cdot pl(\bar{\gamma}_i) \prod_{j=1}^{i-1} (1 - pl(\bar{\gamma}_j))}. \tag{4.24}$$

First, we consider the scenario where N_R is fixed.

Theorem 4.1 *When N_R is constant, the lower bound of energy-delay tradeoff represented by (4.22) and (4.23) is achieved if and only if the SNR vector $\bar{\Gamma}$ is equal to*

$$\bar{\Gamma}_{opt} = \arg \min_{\bar{\Gamma} \in (R^+)^{N_R}} g(\bar{\Gamma}) \tag{4.25}$$

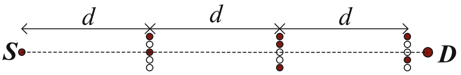
Proof Refer to Appendix A in [22].

The lower bound of energy-delay tradeoff is thus characterized by

$$\overline{EDRb}(\bar{\Gamma}_{opt}, P_t, N_R) = \frac{\overline{DDR}(\bar{\Gamma}_{opt}, P_t, N_R)}{D_{hop}} \cdot (E_c + K_1 P_t), \tag{4.26}$$

where N_R and Γ_{opt} are constant. The priority between energy and delay is balanced with the transmission power P_t.

Fig. 4.2 Approximation solution



From this bound, according to (4.23), the optimal transmission power for a given delay constraint, $\overline{DDR} = ddr$, is derived as

$$P_{opt}(N_R) = \frac{1}{K_2} \cdot \left(g(\bar{\Gamma}_{opt}, N_R) \cdot \frac{D_{hop}}{ddr} \right)^{\alpha}. \tag{4.27}$$

Next, we show how to obtain $\bar{\Gamma}_{opt}$. For a given set of nodes with fixed SNR values, we derive the following theorem from (4.19).

Theorem 4.2 *For a given set of N_R forwarding nodes whose corresponding $\bar{\gamma}$ values are $\bar{\gamma}_1, \ldots, \bar{\gamma}_{N_R}$, respectively $g(\bar{\Gamma}, N_R)$, is minimized with respect to $\bar{\gamma}$ if and only if the $\bar{\gamma}_i$s are ordered in an increasing order such that the higher priority is given to the node with the smaller $\bar{\gamma}_i$.*

Proof Refer to Appendix B in [22].

Then, we have to find the SNR values $\bar{\Gamma}$ that minimize $g(\bar{\Gamma})$, according to Theorem 2. Due to computational complexity and for the sake of simplicity, we assume that all the forwarding nodes are located in the same area and thus that all values of $\bar{\gamma}_i$ are equal to an optimal value $\bar{\gamma}_{opt}$. This means that the N_R receivers have the same effective transmission distance d and are deployed around the line between a source node and a destination node as shown in Fig. 4.2. In this way $(g\bar{\Gamma}_{opt})$, can be approximated by:

$$\tilde{g}(\bar{\gamma}_{opt}) = \frac{\left(1 + \frac{N_R}{90}\right) \cdot \bar{\gamma}_{opt}^{\frac{1}{\alpha}}}{1 - (1 - pl(\bar{\gamma}_{opt}))^{N_R}}. \tag{4.28}$$

Figure 4.3 shows the exact value of minimum $g(\bar{\Gamma}_{opt})$ and its approximation $\tilde{g}(\bar{\Gamma}_{opt})$ under the same condition. Note that the difference between the exact value and the approximation is very small. Therefore, we use $\tilde{g}(\bar{\gamma}_{opt})$ to analyze the lower bound of tradeoff between $\overline{EDRb}$ and $\overline{DDR}$ in the following.

The minimum value of $\tilde{g}(\bar{\gamma})$ is achieved from its first derivative with respect to $(\bar{\gamma})$, denoted as

$$\bar{\gamma}_{opt} = \frac{1 - (1 - pl(\bar{\gamma}_{opt}))^{N_R}}{\alpha \cdot N_R \cdot (1 - pl(\bar{\gamma}_{opt}))^{N_R - 1} \cdot pl'(\bar{\gamma}_{opt})}, \tag{4.29}$$

where $pl'(\cdot)$ is the derivation of $pl(\cdot)$ with respect to $\bar{\gamma}$.

We notice that the whole lower bound of the energy-delay tradeoff is obtained for the same SNR value at the receivers. Furthermore, note that this SNR constraint can be achieved for different couples of transmission power and effective distance

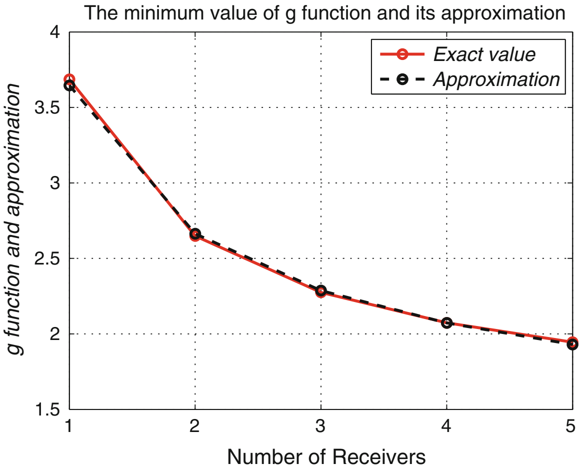


Fig. 4.3 Approximation of g function

parameters. In other words, to satisfy the optimal SNR constraint, the internode distance and the power transmission should be selected jointly according to a desired tradeoff constraint between delay and energy.

Eq. (4.29) implies that the optimal SNR strongly relies on the function $pl(\cdot)$. Thus, we should consider different channels to obtain the closed-form expression of its lower bound. First, we focus on AWGN, Rayleigh fast fading, and Rayleigh block fading channels; then, a general solution for obtaining $\tilde{g}(\bar{\gamma}_{opt})$ is given for all other scenarios.

Substituting (4.12), (4.13), and (4.14) into (4.29), respectively, yields $\bar{\gamma}_{opt}$ in the different kinds of channels.

- AWGN Channel

$$\bar{\gamma}_{optg} = \frac{1}{0.5418\beta_m}\ln\left(0.1826\alpha_m\cdot\left(1-\left(1-\left(-\alpha W_{-1}\left(\frac{(0.1826N_b\alpha_m)^{-N_R}}{-\alpha}\right)\right)^{-1/N_R}\right)^{-1/N_b}\right)^{-1}\right), \tag{4.30}$$

where $W_{-1}[\cdot]$ is the branch of the Lambert W function satisfying $W(x) < -1$ [23].
- Rayleigh Block Fading Channel

$$\bar{\gamma}_{optb} = \frac{-4.25\log_{10}(N_b)+2.2}{\beta_m\ln\left(1-(1/(\alpha N_R))^{1/N_R}\right)}. \tag{4.31}$$

- Rayleigh Fast Fading Channel

$$\bar{\gamma}_{optf} = \frac{\alpha_m}{2\beta_m\left(1-\left(1-(1+\alpha N_R)^{-1/N_R}\right)^{1/N_b}\right)}. \tag{4.32}$$

Substituting $\bar{\gamma}_{optg}$, $\bar{\gamma}_{optb}$ and $\bar{\gamma}_{optf}$ into (4.28), respectively, we can achieve $\tilde{g}(\bar{\gamma}_{opt})$ in the aforementioned three kinds of channels.

Contrarily to the cases mentioned above, the closed-form expression of $\bar{\gamma}_{opt}$ or $\tilde{g}(\bar{\gamma}_{opt})$ in many practical situations cannot be derived since the expression of link probability is unknown. A numeric approach is thus employed to obtain the approximated value of $\bar{\gamma}_{opt}$ or $\tilde{g}(\bar{\gamma}_{opt})$ by using (4.28).

4.3.2 Optimal Number of Receivers

In the previous subsection, the lower bound of energy delay tradeoff is obtained for a fixed number of receivers. In this subsection, we analyze how to select the optimal number of receivers for a given *ddr*.

Theorem 4.3 $\overline{BDRb}(\bar{\Gamma}, P_t, N_R)$ *is convex with respect to* N_R.

Proof Refer to Appendix C in [22].

Theorem 3 proves that a global minimum exists. In addition, since the optimal power P_{opt} is known explicitly with (4.27) under a delay constraint *ddr*, the minimum value of $\overline{EDRb}$ for a given delay constraint *ddr* can be obtained by searching the optimal number of receivers. The following algorithm addresses this optimization.

Algorithm 1 Search the optimal number of receivers N_{Ropt}.

```
$N_R \leftarrow 1$ $\overline{\mathrm{EDRb1}} \leftarrow \mathrm{inf}$, flag $\leftarrow 0$, ddr
while flag $==0$ do
    Calculate $\tilde{g}(\bar{\gamma}_{opt}, N_R)$
    $\overline{\mathrm{EDRb}} \leftarrow E_c(N_R) \cdot \frac{\mathrm{ddr}}{D_{\mathrm{hop}}} + \tilde{g}(\bar{\gamma}_{opt}, N_R)^{\alpha} \cdot \frac{K_1}{K_2} \cdot \left(\frac{\mathrm{ddr}}{D_{\mathrm{hop}}}\right)^{1-\alpha}$
    if $\overline{\mathrm{EDRb}} > \overline{\mathrm{EDRb1}}$ then
        flag $\leftarrow 1$, $N_{Ropt} \leftarrow N_R - 1$
        return $N_{Ropt}$
    else
        $\overline{\mathrm{EDRb}} \leftarrow \overline{\mathrm{EDRb}}$, $N_R \leftarrow N_R + 1$
    end if
end while
```

4.3.3 Lower Bound of Energy-Delay Tradeoff

The proposed algorithm offers a method to achieve N_{Ropt} under a delay constraint *ddr*. Then, substituting N_{Ropt} into (4.26) and (4.27), the lower bound of energy-delay tradeoff for one-hop transmission with opportunistic routing is

$$\overline{EDRb}_{opt} = (E_c(N_{Ropt}) + K_1 \cdot P_{opt}(N_{Ropt})) \cdot \frac{ddr}{D_{hop}}. \tag{4.33}$$

4.3.4 *Minimum Energy Consumption*

In the previous section, we found the lowest point existing in each curve of lower bound of energy-delay tradeoff in three kinds of channels. For delay-tolerant applications, the minimum energy consumption point is very important, that is, the lowest point on the curve of lower bound of energy-delay tradeoff. In the following subsection, we will derive this point.

In this subsection, as to the lowest point, we derive the lower bound of energy efficiency and corresponding energy-optimal transmission power and distance without the delay constraints.

4.3.4.1 Optimal Transmission Power

Assuming N_R is constant, in order to get the minimum value of $\overline{EDRb}$, it is obvious that we should minimize $\tilde{g}(\bar{\gamma})$ and $f(P_t) = (E_c(N_R) + K_1 P_t)/(K_2 P_t)^{1/\alpha}$ at the same time in (4.22). Letting $\bar{\gamma} = \bar{\gamma}_{opt}$, we have $\tilde{g}(\bar{\gamma}_{opt})$. Then, employing Lagrange algorithm, we obtain

$$\frac{d}{dP_t}\left(\frac{E_c(N_R) + K_1 P_t}{(K_2 P_t)^{1/\alpha}}\right) = 0.$$

Solving the above equation yields

$$P_0 = \frac{E_c(N_R)}{(\alpha - 1) \cdot K_1}. \tag{4.34}$$

Substituting (4.8) and (4.9) into (4.34) yields

$$P_0(N_R) = \frac{(N_R + 1) \cdot T_{start} \cdot P_{start}}{\beta_{amp} \cdot (\alpha - 1) \cdot \frac{N_b}{R_b R_{code}}} + \frac{N_R P_{rxElec} + P_{txElec}}{\beta_{amp} \cdot (\alpha - 1)}, \tag{4.35}$$

where $(N_b + N_{head})/(R_b R_{code})$ is the transmission duration of a packet. Since $(N_b + N_{head})/(R_b R_{code}) \gg T_{start}$ in the general case, the first part of (4.35) can be neglected. Thus, we get

$$P_0 \approx \frac{N_R P_{rxElec} + P_{txElec}}{\beta_{amp}} \cdot (\alpha - 1). \tag{4.36}$$

It should be noted that p_0 is tightly related to N_R, so that we should apply Algorithm 1 to find the optimal number of receivers N_{R0} which is closely relevant to the modulation and to the type of channel of a network. Then, substituting

N_{R0} into (4.36) yields

$$P_0 \approx \frac{N_{R0} P_{rxElec} + P_{txElec}}{\beta_{amp} \cdot (\alpha - 1)}. \tag{4.37}$$

We note that p_0 is independent of τ_{ack}.

4.3.4.2 Lower Bound of $\overline{EDRb}$ and Its Corresponding Delay

The lower bound of $\overline{EDRb}$ is obtained by substituting (4.37) into (4.22):

$$\overline{EDRb_0} = \frac{E_c(N_{R0}) + K_1 P_0}{(K_2 P_0)^{1/\alpha}} \cdot \tilde{g}(\bar{\gamma}_{opt}, N_{R0}). \tag{4.38}$$

Based on this result, we can set the transmission power of node according to (4.36) to minimize the total energy consumption for the delay-tolerant applications. Note that this value is the threshold of transmission power, this is to say, the transmission power of nodes should not be smaller than this value; otherwise, a network system will be running in an inefficient state as shown on the right side curve of the lowest point in Fig. 4.4.

Moreover, on the basis of (4.23) and (4.37), the corresponding delay of the state of minimum energy consumption, that is, maximal mean delay $\bar{D}_{max}$ is obtained by

$$\bar{D}_{max} = D_{hop} \cdot \frac{d_{SD}}{(K_2 P_0)^{1/\alpha}} \cdot \tilde{g}(\bar{\gamma}_{opt}, N_{R0}). \tag{4.39}$$

where d_{SD} is the distance between a source node and a destination node.

4.3.4.3 Minimum Mean Transmission Distance

Base on (4.40) and (4.11), we obtain the minimal mean transmission distance:

$$d_0 = \left(\frac{K_2 P_0}{\bar{\gamma}_{opt}}\right)^{\frac{1}{\alpha}}. \tag{4.40}$$

This distance shows the minimal distance between a source node and its relay nodes or its destination node for one-hop transmissions; otherwise, too small hop distance results in more energy consumption or too many hops, namely, too much delay.

According to (4.37), (4.42), (4.43), when the transmission environment deteriorates, P_0 reduces with the increase of α. This fact results in the decrease of transmission distance per hop. Accordingly, a larger number of hops is required to achieve the transmission between a certain pair of source and destination nodes, which ultimately brings a greater delay and overall energy consumption. On the other side, in terms of the efficient amplifiers that correspond to smaller values of β_{amp}, P_0

should be augmented in order to reach a larger transmission distance at each hop. Consequently, the related energy expenditure and delay can be decreased.

4.4 Energy-Latency Tradeoff of Multi-Hop Transmissions

In this section, we extend the result of the one-hop transmission case developed in Sect. 4.3 to the scenarios of multi-hop transmission. Meanwhile, the effect of physical parameters on lower bound of energy-latency tradeoff and the energy efficiency gain of opportunistic routing are analyzed.

4.4.1 Lower Bound of Energy-Latency Tradeoff

The lower bound of energy-latency tradeoff can be abstracted as an optimization problem:

$$\begin{aligned} &\text{Minimize: } \bar{E}_{tot} \\ &\text{Subject to: } \bar{D}_{tot} \le Delay\ limit. \end{aligned}$$

Here, $\bar{E}_{tot}$ and $\bar{D}_{tot}$ are the end-to-end energy consumption and latency between a source node and its destination.

In order to obtain the lower bound of energy-latency tradeoff of multi-hop transmission, the theorems about *equivalent distance transmission* are introduced in the following.

Theorem 4.4 *In a homogeneous network, a source $\mathcal{S}$ node sends a packet of N_b bits to a destination $\mathcal{D}$ using n hops in opportunistic communication mode. The n relaying clusters are located around $\mathcal{S}$–$\mathcal{D}$ line, as shown in Fig. 4.1, and each cluster has the same number of relay candidates (N_R). The distance between $\mathcal{S}$ and $\mathcal{D}$ is $d_{\mathcal{SD}}$. The length of each hop is $d_1, d_2, \ldots, d_n$, respectively, and the average EDRb is denoted as $\overline{EDRb}(d)$. The minimum mean total energy consumption $\overline{Etot}_{min}$ is obtained if and only if $d_1 = d_2 = \ldots = d_n$.*

$$\overline{Etot}_{min} = N_b \cdot \overline{EDRb}(d_{\mathcal{SD}}/n) \cdot d_{\mathcal{SD}}. \tag{4.41}$$

Proof Refer to Appendix D in [22].

Theorem 4.5 *On the same assumption as Theorem 4, the mean delay of one-hop transmission is referred to as $\bar{D}(d)$. The minimum mean end-to-end delay $\overline{Dtot}_{min}$ is given by*

$$\overline{Dtot}_{min} = \bar{D}(d_{\mathcal{SD}}/n) \cdot n. \tag{4.42}$$

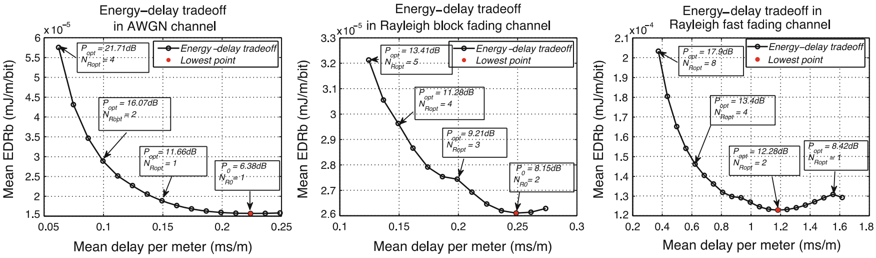


Fig. 4.4 Lower bound of $\overline{EDRb}-\overline{DDR}$ tradeoff and corresponding parameters in different channels

if and only if $d_1 = d_2 = \ldots = d_n$.

Proof Refer to Appendix E in [22].

Based on Theorems 4 and 5, we conclude that regarding a pair of source and destination nodes with a given number of hops, the single scenario, which minimizes both mean energy consumption and mean transmission delay, is that each hop with uniform distance along a linear path. As a result, the optimization about energy and delay for a single hop will bring the optimization of the same performance for the multi-hop transmission. Hence, the results about the lower bound of energy-delay tradeoff in Sect. 4.3 can be employed directly in the scenario of multi-hop transmissions.

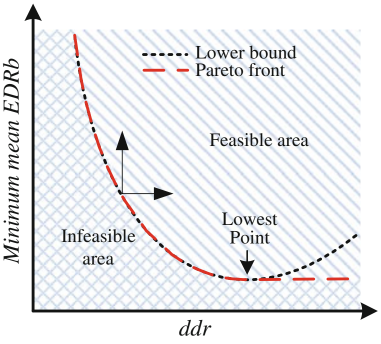
4.4.2 Energy-Latency Tradeoff in Different Channels

According to the analysis in previous subsections, we obtain P_{opt} and N_{Ropt} under a delay constraint *ddr*, then we have the lower bound of energy-latency tradeoff for multi-hop transmission with opportunistic routing on the basis of

tightly depends on the function of link probability $pl(\cdot)$, we analyze the lower bound of energy-latency tradeoff in three different channels mentioned in this subsection.

Figure 4.4 shows the lower bound of energy-latency tradeoff, the corresponding optimal transmission power and the optimal number of receivers in different channels. It should be noted that the lowest point exists in each curve; this is to say, there is the most energy saving point without the delay constraints for each channel, and the corresponding mean delay is the maximum mean delay of a pair of nodes. In Sect. 4.3.4, we have analyzed the most energy saving point in detail. The left side of the lowest point shows the energy consumption increases with the decrease of the delay constraint, which coincides with our intuition. However, in the right of the lowest point, the energy consumption increases with the increase of the delay because both the transmission power and the number of receivers are too small which results

Fig. 4.5 Lower bound and Pareto front of the energy-latency tradeoff



in very small one-hop distance, that is, the increase of the hop number. Certainly, this work state should be avoided in practice.

Note that the optimal number of receivers corresponding to the lowest point in each curve is 2 in Rayleigh block fading and fast fading channels and is 1 in an AWGN channel. The result implies that too many nodes will lead to the waste of energy; this is to say, we should avoid acting all neighbor nodes as the relay candidates. In addition, the optimal number of receivers rises with the decrease of delay limit. As for the corresponding optimal transmission power, it is not monotonically decreased as we saw in traditional multi-hop communications. Conclusively, it is clear that the transmission power and the number of relay candidates should be adjusted correctly according to a delay constraint in order to avoid too much energy consumption. Algorithm 1 and (4.27) provide the approach to calculate the optimal transmission power and a distributed algorithm to select the optimal relay candidates will be introduced in Sect. 4.5.

4.4.3 Pareto Front of Energy-Latency Tradeoff

It is rationally deduced that $\overline{EDRb}$ is a convex function with respect to *ddr*, as shown in Fig. 4.4. Besides, a lowest point corresponding the minimum energy consumption point exists in the curve of $\overline{EDRb}-\overline{DDR}$. The curve on the left side of the lowest point is exactly the Pareto front of the energy-latency tradeoff described in Fig. 4.5. In other words, the Pareto front is a subset of the lower bound of energy-latency tradeoff.

This Pareto front separates the $\overline{EDRb}-\overline{DDR}$ area into two parts: feasible area and infeasible area as shown in Fig. 4.5. That is to say, the point corresponding to the energy-latency state of a network could exist in the feasible area, but is not possible to access the infeasible area. Hence, in order to obtain the Pareto front of the energy-latency tradeoff, we need not only the lower bound of energy-latency tradeoff, but also the lowest point in the lower bound.

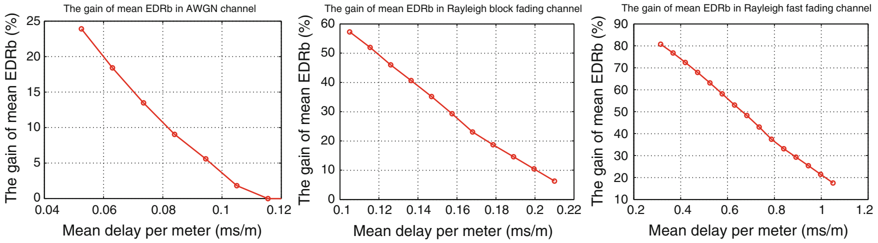


Fig. 4.6 Comparisons between traditional multi-hop routing and opportunistic routing

4.4.4 The Gain of Opportunistic Routing

By the Pareto front of the energy-latency tradeoff, we analyze the energy efficiency gain of opportunistic routing compared with traditional multi-hop routing in this subsection. The benefit of opportunistic routing in terms of energy efficiency under the same delay constraint, as shown in Fig. 4.6, is measured with the energy gain defined as

$$Gain = \frac{\overline{EDRb}_{p2p} - \overline{EDRb}_{opp}}{\overline{EDRb}_{p2p}}, \tag{4.43}$$

where the $\overline{EDRb}_{p2p}$ is the optimal $\overline{EDRb}$ with a delay constraint ddr in traditional point-to-point multi-hop communications, which is obtained according to the approach proposed in [17], and $\overline{EDRb}_{opp}$ is referred to as the optimal $\overline{EDRb}$ with the same delay constraint using opportunistic routing obtained by (4.33).

Figure 4.6 provides an example in three kinds of channel and the physical parameters are shown in Table 4.1. In this example, the gain of opportunistic routing decreases from 25 to 0 % with the increase of the delay constraint. The gain becomes 0 when the delay constraint is greater than 0.11 ms/m, which implies that the opportunistic routing has changed to the traditional multi-hop routing. This result coincides with the result in Fig. 4.4, where the optimal number of receiver becomes 1 for the corresponding delay constraint. In other words, when the delay constraint is greater than a threshold, the traditional multi-hop communication is more energy efficient than the opportunistic routing in an AWGN channel.

In the Rayleigh block fading and Rayleigh fast fading channels, the gain of opportunistic routing is always greater than 0, which reveals opportunistic routing outperform the traditional multi-hop routing in these two kinds of channel, where opportunistic routing benefits from the effect of diversity and can improve the energy efficiency, while the gain decreases with the increase of the delay constraint.

According to these results, it can be concluded that opportunistic routing is more energy efficient for Rayleigh block fading channels than for AWGN channels.

4.5 Simulations

The purpose of this section is to verify the theoretical analysis of the lower bound on the energy-latency tradeoff and the energy efficiency in a two-dimension Poisson distributed network by simulations although these theoretical results are obtained in a linear network using approximation approach. First, we introduce a novel opportunistic protocol on the basis of the theoretical analysis.

4.5.1 Opportunistic Protocol

The analysis in Sect. 4.3 reveals that the transmission power and the number of receivers should be configured as the corresponding optimal values in order to approach the Pareto front of energy-latency tradeoff. Nevertheless, Algorithm 1 proposed in Sect. 4.3 requires the acknowledgment of global network parameters, which cannot be directly applied to the distributed networks. Consequently, a distributed algorithm is introduced based on the algorithm proposed in [4] and the analysis in Sect. 4.3.

The containing property in Lemma 3.4 proposed in [4] shows that a straightforward way to find an optimal set containing r nodes is to add a new node into the optimal node set containing $r-1$ nodes. Furthermore, when a local minimum $\overline{EDRb}$ is found, it is the global minimum according to Theorem 3. Based on this idea, a distributed algorithm for finding the optimal receiver set at each hop in order to minimize the energy consumption and satisfy the delay constraint *ddr* is proposed in Algorithm 2.

Algorithm 2 Search the optimal set of receivers

```
$B \leftarrow C$, $F^* \leftarrow F_c^* \leftarrow F \leftarrow \varnothing$, ddr
while $C \neq \phi$ do
    for each node $i \in B$ do
        $F^* \leftarrow F \cup \{i\}$;
        Sorting the nodes in $F^*$
        according to their effective transmission distance;
        ddr$^* \leftarrow \overline{\text{DDR}}(F^*)$
        if ddr$^* > ddr_c^*$ then
            ddr$_c^* \leftarrow ddr^*$; $F_c^* = F^*$
        end if
    end for
    if ddr$_c^* < ddr$ then
        $F \leftarrow F_c^*$; $B \leftarrow C \leftarrow F$
    else
        return $F_c^*$
    end if
```

In Algorithm 2, C is the set of neighbor nodes of a source node, F is the set of nodes selected to receive the packet from the source node.

Next, we introduce the process of this protocol.

(1) Search the forward candidates according to Algorithm 2.
(2) Assign a priority to each node according to its effective transmission distance.
(3) Transmit the data packet including the information of relay candidates ID and their corresponding priorities.
(4) Nodes in the set of relay candidates try to receiver packet.
(5) A node that receives the packet correctly calculates the backoff time according to the priority and waits for the ACK packet from the nodes with the priority higher than that of itself.
(6) If a node does not receive any ACK packet, it broadcasts its ACK packet and then is ready to transmit the received packet to next hop or destination. If a node receives an ACK packet, the received data packet is dropped from its queue.
(7) The source node waits for the ACK packet from one of forwarding candidates. If an ACK packet is received, the source node removes the packet from the buffer; otherwise, it is ready to retransmit the data packet.

4.5.2 Simulation Setup

In the simulations, the lower bounds of energy-latency tradeoff and $\overline{EDRb}$ are evaluated in an area $\mathcal{A}$ with surface $S_A = 100 \times 1200\ \text{m}^2$ using the simulator WSNet [24]. The nodes are uniquely deployed according to Poisson distribution.

All the other simulation parameters concerning a node are listed in Table 4.1. The distance between the source and the destination nodes is 1,000 m. The source node transmits only one data packet of 320 bytes to the destination with BPSK modulation. The size of ACK packet is 26 bytes. For every hop, the transmitter will retransmit the data packet until the data packet is received by the next relay node; this is to say, there is no limit for the number of retransmissions in order to ensure the reliability. The opportunistic protocol proposed in Sect. 4.5.1 is employed to simulations. A simulation will be repeated for 1,000 times in each different configuration.

The network model used in the simulations assumes the following statements.

(i) After the initial phase, the network is geographically aware; that is, each node knows the position of itself, the sink node, and all the neighbor nodes in the network.
(ii) Each node in the network has the same fixed transmission power.

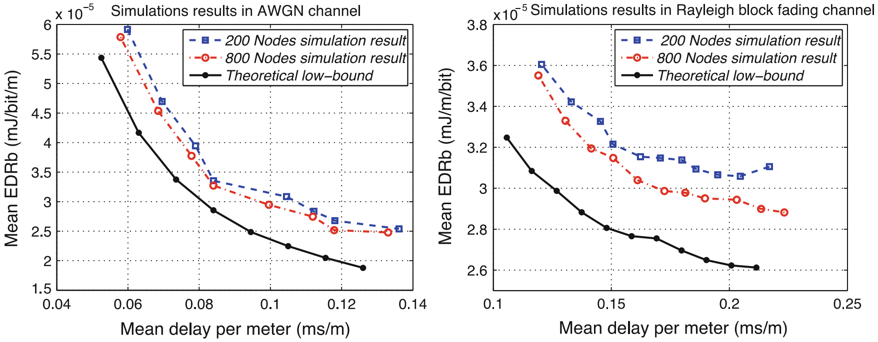


Fig. 4.7 Simulation results of $\overline{EDRb}-\overline{DDR}$ tradeoff in different node densities

4.5.3 *Results and Analysis*

In order to analyze the variety of energy-latency tradeoff with the increase of node density, the simulations are run in three cases: 200 and 800 nodes in the simulation area. The transmission power of each node is configured as the optimal transmission power derived from (4.27).

Figure 4.7 provides the simulation results which are compared with the theoretical lower bound of energy-latency tradeoff in AWGN and Rayleigh block fading channels. The comparison reveals that the simulation results approach to the theoretical lower bound with the increase of the node density, because the distance between pairwise relay nodes achieved from the routing

scheme is more and more near the optimal transmission distance at each hop when the node density increases. And we can deduce that the lower bound can be reached when the node density is high enough. Hence, it is concluded that the theoretical lower bound on $\overline{EDRb}$ is adequate for a two-dimension Poisson network although its derivation is based on a linear network using an approximation method. Furthermore, these results imply that unreliable links play an important role in energy saving.

4.6 Conclusions

We note that the closed-form expression of the lower bound of the energy-latency tradeoff in (4.26) includes all parameters of the physical, MAC and routing layers. Therefore, (4.26) provides a cross-layer framework to both evaluate the energy-latency performance and optimize these parameters. This framework can be used in the following applications:

(1) Performance evaluation

During the design or the planning phase of a network, these results about the performance evaluation provide the basis for the choice of sensor nodes and for the choice of routing and MAC layer protocols.

(2) Benchmark of performance
Regarding the design of a protocol, the best performance of a network obtained by this framework can act as the benchmark performance in order to measure the performance of a protocol and to adjust its parameters.

(3) Parameter optimization
We can optimize parameters such as the transmission power according to the request of performance of a network on the basis of the framework.

In this chapter, an unreliable link model is firstly integrated into our energy model using the specific metrics for energy efficiency and delay: $\overline{EDRb}$, $\overline{DDR}$. By minimizing $\overline{EDRb}$ for AWGN, Rayleigh block fading and Rayleigh fast fading channels with and without delay constraint, we have shown that the channel state impacts the optimal number of receivers in a cluster. Meanwhile, the corresponding optimal transmission power and the optimal transmission range are obtained. The Pareto fronts of energy-latency tradeoff for one-hop and multi-hop transmissions are analyzed and compared with that of traditional multi-hop communications. The main conclusion is that opportunistic routing exploiting spatial diversity is beneficial for the Rayleigh fast fading and the Rayleigh block fading channels.

References

1. Pelusi L, Passarella A, Conti M (2006) Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. IEEE Commun Mag 44(11):134–141
2. Zorzi M, Rao RR (2003) Geographic random forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance. IEEE Trans Mob Comput 2(4):349–365
3. Biswas S, Morris R (2005) EXOR: opportunistic multi-hop routing for wireless networks. In: Proceedings of ACM SIGCOMM, pp 133–144
4. Zeng K et al (2007) On geographic collaborative forwarding in wireless ad hoc and sensor networks. In: Proceedings of WASA, pp 11–18
5. Jaffrès-Runser K, Gorce J-M, Ubéda S (2006) Multiobjective QoS-oriented planning for indoor wireless LANs, In: Proceedings of IEEE VTC, pp 1–5
6. Shah RC, Wiethölter S, Wolisz A, Rabaey JM (2005) Modeling and analysis of opportunistic routing in low traffic scenarios. In: Proceedings of WiOpt, pp 294–304
7. Luk C-P, Lau W-C, Yue O-C (2008) An analysis of opportunistic routing in wireless mesh network. In: Proceedings of IEEE ICC, pp 2877–2883
8. Gorce J-M, Zhang R, Parvery H (2007) Impact of radio link unreliability on the connectivity of wireless sensor networks. EURASIP J Wireless Commun Networking 2007:1–16
9. Ganesan D et al (2003) Complex behavior at scale: an experimental study of low-power wireless sensor networks, UCLA, Technical Report CS TR 02–0013
10. Woo A, Culler DE (2003) Evaluation of efficient link reliability estimators for low-power wireless networks. Technical Report, Computer Science Division, University of California
11. Zhao J, Govindan R (2003) Understanding packet delivery performance in dense wireless networks. In: Proceedings of ACM SenSys, pp 1–13
12. Zúñiga Zamalloa M, Krishnamachari B (2007) An analysis of unreliability and asymmetry in low-power wireless links. ACM Trans Sens Netw 3(2):1–34

13. Shah RC et al (2005) When does opportunistic routing make sense? In: Proceedings of IEEE PerCom, pp 350–356
14. Zhang L, Soong B-H (2008) Energy efficiency analysis of channel aware geographic-informed forwarding (CAGIF) for wireless sensor networks. IEEE Trans Wireless Commun 7(6): 2033–2038
15. Zhang R et al (2009) Energy efficiency of opportunistic routing with unreliable links. In: Proceedings of IEEE WCNC
16. Micaz datasheet, http://www.xbow.com
17. Karl H, Willig A (2005) Protocols and architectures for wireless sensor networks. John Wiley and Sons, New York, USA
18. Zhang R et al (2008) Energy-delay bounds analysis in wireless multi-hop networks with unreliable radio links, ARES/INRIA, Technical Report 6598
19. Cerpa A et al (2005) Statistical model of lossy links in wireless sensor networks. In: Proceedings of IEEE IPSN
20. Gao JL (2002) Analysis of energy consumption for ad hoc wireless sensor networks using a bit-meter-per-joule metric. Technical Report 42–150, Jet Propulsion Laboratory, California Institute of Technology
21. Li D, Sun X (2006) Nonlinear integer programming. Springer, New York, USA
22. Zhang R, Gorce J-M, Berder O, Sentieys O (2011) Lower Bound of Energy-Latency Tradeoff of Opportunistic Routing in Multihop Networks. EURASIP J Wireless Commun Networking ID 265083:1–17
23. Corless RM et al (1996) On the lambertw function. Adv Comput Math 5(4):329–359
24. http://wsnet.gforge.inria.fr/

Chapter 5
Routing Protocols in Infrastructure-Based Opportunistic Networks

Sanjay K. Dhurandher, Deepak Kumar Sharma, Isaac Woungang, Sahil Gupta and Sourabh Goyal

Abstract This chapter analyzes the infrastructure needed for communication in Opportunistic Networks (OppNets) and the routing protocols for message passing in order to optimize message delivery delays, energy consumption, buffer occupancy, bandwidth requirements, and throughput. Infrastructures used in OppNets can be classified as fixed infrastructure such as infostations which provide high bandwidth connectivity within a specified area and mobile infrastructures such as Message Ferry (MF) which provide a reliable source for communication and transmission of messages between the nodes. Message ferrying scheme proves to be good for sparse networks where network partition problem is very common like in wireless sensor networks (WSNs) due to limited energy and in Mobile Ad-Hoc Networks (MANETs) due to mobility and short radio communication range of nodes. Communication in MF scheme can be from node to ferry and ferry to node or even from ferry to ferry. Their main aim is to reduce delays in message delivery, proper management of message buffers, and optimization of energy consumption. Designing the route

---

S. K. Dhurandher (✉)
CAITFS, Division of Information Technology, Netaji Subhas Institute of Technology,
University of Delhi, New Delhi, India
e-mail: dhurandher@gmail.com

D. K. Sharma · S. Gupta · S. Goyal
Division of Computer Engineering, Netaji Subhas Institute of Technology,
University of Delhi, New Delhi, India
e-mail: dk.sharma1982@yahoo.com

S. Gupta
e-mail: sahilgupta221@gmail.com

S. Goyal
e-mail: sourabh912@gmail.com

I. Woungang
Department of Computer Science, Ryerson University, Toronto, Canada
e-mail: iwoungan@scs.ryerson.ca

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_5,

of a message ferry is a NP hard problem. In case of ferry failure, designing ferry replacement schemes is also an interesting issue to be explored.

Keywords Opportunistic networks · Delay tolerant networks · Infostation · Message ferry · Infrastructure-based protocols · Wireless sensor networks · Network partition · Shared wireless infostation model (SWIM) · Mobile ubiquitous LAN extensions (MULEs).

5.1 Introduction

In the recent years, Opportunistic Networks [1] have gained a very large popularity among the research community. OppNets are emerging as an important means of communication between heterogeneous devices due to their portability and flexibility of operation/computation. Traditional ad hoc networks and Internet routing protocols route the message by selecting the best path toward the destination. Once a path has been established between sender and receiver, they assume it to be fixed for the entire duration of message transfer. Thus, these protocols do not work well in case of OppNets where there is never a fixed path between sender and receiver due to node's mobility, network partitions, node failure, etc. [2, 3].

The routing in OppNets is based on the contact opportunity between the nodes that arises due to their mobility, and the store carry and forward technique. Route building in OppNets is not static but dynamic as the source node can select any node as next hop from a group of its neighbors assuming it to be the best forwarder toward the destination. The message is passed to the intermediate nodes which takes it closer and closer to the destination and finally to the destination itself. If the intermediate node does not find a suitable node which promises to take the message closer to the destination, it keeps the message copy in its buffer until it finds a suitable node or finds the destination node itself [4, 5]. Nodes are also required to have adequate amount of buffer to store all the messages until they are transferred to the next suitable node. So messages in this type of networks can suffer long delays and this is the reason why they are considered as the sub-class of *Delay Tolerant Networks* (DTNs) [6–8].

The routing and forwarding protocols in OppNets can be classified into two categories—infrastructure-less and infrastructure based [9].

(1) *Infrastructure-less protocols* These protocols have been designed for flat ad hoc networks. They make use of the contact opportunity between the nodes and their mobility to forward the message. Messages can be transferred by nodes themselves with suitable techniques. Routing and forwarding every node in this type of network is similar in nature and no categorization in terms of priority, hierarchy, etc., is assigned to the nodes in order to distinguish between them. Examples of such type of protocols are Epidemic Routing [10], Spray and Wait [11], Spray and Focus [12], Prophet [13], HiBOp [14] etc.

(2) *Infrastructure-based protocols* These protocols exploit some form of infrastructure to opportunistically forward the messages to the destination. The infrastructure is generally composed of some special nodes that are more powerful in terms of energy, transmission range, buffer space, etc., as compared to the normal nodes. The base stations and access points are often involved in the routing and forwarding mechanism. Some reliable agents are used in the form of infrastructure for carrying message toward the destination. These agents are held responsible for efficient message delivery with minimum energy consumption, buffer occupancy, delivery delays and maximum security, delivery ratio etc. These agents generally include infostations [15], message ferries [16], and data MULEs [17]. Infrastructure-based routing protocols in OppNets can be further classified into fixed infrastructure and mobile infrastructure, depending upon the type of infrastructure they use in the network [9]. Nodes of the fixed infrastructure are kept fixed at some geographical locations (e.g., Infostations), while nodes of mobile infrastructure can move around in the network following either a fixed predetermined path or a completely random path (e.g., message ferries).

In this chapter, the main focus will be only on the routing protocols for infrastructure-based OppNets and their impact on the whole network. The rest of the chapter is organized as follows. In Sect. 5.2, a detailed overview of the related work in the area of routing protocols for infrastructure-based OppNets is presented. Section 5.3 gives the final conclusive remarks related to the chapter.

5.2 Background and Related Work

In this section, a brief literature survey of the significant work done in the area of routing protocols for infrastructure-based opportunistic networks is presented. The routing protocols have been classified into different categories depending upon the type of infrastructure they employ for message transfer. The classification is done in four sections namely infrastructure in form of infostations, infrastructure in form of message ferries, infrastructure in form of access points and data MULEs, and infrastructure in form of buses and roadside units.

5.2.1 Infrastructure in Form of Infostations

Infostations [15] are an array of isolated objects of high bandwidth connectivity that provide high bit rate services in core locations using separate radio and channel. They can share a common channel and provide transmission rate higher than megabits per second if properly spatially separated. They can be installed along roadways, in buildings, at airports, and other known areas [18]. They allow significant amount of information to be transferred in few seconds like fax, E-mail, large data files etc. Individual infostations on highways and airports can provide special gateway to Internet and access to remote servers for messages transfer. Vehicle on the move gets the required data packets from these infostations. There are two types of infostations namely static infostation and mobile infostation [15, 18]. These two differ in terms of network coverage, storage capacity, and throughput. Mobile infostation has higher values of the above-mentioned parameters as compared to the static infostation. All the roadways buses, airplanes etc., can act as mobile infostations.

In [15] Goodman et al. discussed about the static infostation model. In this model, the node that wishes to send a message moves closer to the infostation and uploads the message to it. The infostation keeps the message in its buffer. It is now the responsibility of the infostation to deliver the message to the destination node. It transfers the message to the destination node as soon as it comes in contact of the infostation.

In [18] Cavalcanti et al. introduced the concept of mobile infostation. In this model, the source node gives the message packet to mobile infostation which carries the message along with it till it finds suitable nodes which can take the message closer to the destination or to the destination node itself. As the mobile infostations move along with the users, the network coverage also improves as nodes remain in contact with the infostation for a longer period of time. The improvement in the coverage capacity depends on the direction of motion of the user and the mobile infostation. If the user is moving in the direction of the mobile infostation then it remains in contact for a longer period of time as compared to the user moving in the direction opposite to that of the mobile infostation. The proposed system of mobile infostation in [18] is not suitable for real-time applications as the implementation of mobile infostation is not feasible.

In [19] Small et al. extended and integrated the concept of infostation with the ad hoc networks and this new model is called *Shared Wireless Infostation Model (SWIM)*. In this model, whenever two nodes come in contact with each other, they exchange the information they are carrying with them. When any node comes in contact with the infostation it offloads all its information to it and the memory of that node is cleared. Every node maintains the information of all the delivered packets and the ones it is carrying currently, so that it does not accept the same packet in future. *SWIM* is studied through the biological information acquisition system for monitoring the whales. Whales are tagged and they act as regular nodes in the system. The offloading stations called *SWIM stations* [19] (where whales offload all the information they are carrying) are placed on the sea surface. The transfer of data occurs in terms of the packets as the whales come on the surface for a very short period of time. So the data transfer speed should be high enough to offload all the information. The range and capacity of the infostations should be kept high. One of the challenging issues in this model is the proper placement of the infostations, so that delay can be minimized. Other issue is about the size of the memory of tags that the whales are carrying with them. To resolve the storage issue, *Markov Chains* [20] (used for modeling infectious disease) is used and with a certain probability level time-to-live for a packet is set such that it is delivered to one of the infostations before the packet is discarded. Contact rate (rate with which whales come in contact) and recovery rate (rate with which a whale comes in contact to the infostation) are taken into consideration to calculate the time-to-live for a message packet. Mathematical details of the model are given in [19].

The contact rate and recovery rate parameters are strongly affected by the mobility model. Three mobility models have been studied in [19]. First model is *Random midway mobility model*. In this model, whales move with a constant *velocity* (not speed) for a small time interval after which its velocity changes. The simulation results show that the recovery rate is directly proportional to the number of infostations and there is no impact on the contact rate. The second model is *Group Feeding Mobility model*. Female whales tend to group together and males tend to group with females but not with other males [21–23]. Direction vector of a female is calculated as the weighted sum of directions of migrations, the directions to the nearest female, and the direction of the nearest feeding area [19]. From this model, it is found that if the infostations are placed at the center of the feeding areas the recovery rate is improved thus reducing the delay. Third model is *Multitier mobility model*. In this model, the *SWIM stations* are placed on the seabirds which are moving over the sea. If the speed of these birds is high as compared to whales then mobile infostations (seabirds) will come in contact with the group of whales more frequently, and thus the packets are offloaded frequently which will reduce the time-to-live for each packet. Thus, the proper positioning of the *SWIM stations* can reduce the time-to-live for the packet and therefore reducing the delay.

5.2.2 Infrastructure in Form of Message Ferry

Due to the intermittent connections between the nodes, the existence of a continuous end-to-end path between the sender and the receiver is not guaranteed in OppNets. So it is desirable to employ a special node called ferry, which moves on a specific path in the network. This ferry node forwards the data for disconnected nodes in order to provide the communication opportunities between them. The ferry node has high energy, buffer, and speed as compared to the other nodes in the network. One of the key challenges in message ferry scheme is to design the ferry route in such a way that can improve certain network characteristics like average data delivery delay, message delivery rate, and data loss ratio.

In [24] Zhao et al. proposed a message ferrying scheme for message transfer in highly partitioned wireless networks. In this scheme, a set of nodes called *message ferries* take up the responsibility for carrying messages between disconnected nodes. The MFs act as relays to provide communication between the disconnected nodes. They move around the deployed area according to known routes and communicate with other nodes they meet on the way. With the knowledge about the ferry routes, nodes can change their trajectories to meet the ferries for transmitting or receiving the messages. If a source node wants to send a message to a destination node, then it gives this message to the ferry which will then deliver it to the destination. Thus with proactive movement of the ferries and the nodes, this scheme provides regular connectivity in an ad hoc networks. This work discusses the following five dimensions to describe the context of message ferrying scheme [24]:

1. Ferry mobility can exist for two scenarios namely messaging and nonmessaging. In messaging, the ferry mobility is designed to optimize the efficiency of messaging. In case of nonmessaging mobility, the ferry route is dependent on various other factors like density of nodes and the path selected by them.
2. Regular nodes can be stationary or mobile and their mobility can also exist for messaging and nonmessaging.
3. Multiple ferries with different capabilities can be employed in the system. Ferries may operate completely independent of each other or their movements may be coordinated for specific purpose.
4. Regular nodes can either work independently or they can form cluster to send and receive data from the ferry. In cluster, one or more nodes can act as a monitor node to communicate with the ferry. The other nodes who wish to communicate with ferry will have to send their information to the monitor node.
5. Ferry can be implemented by employing a special capability node or a regular node can be promoted to perform the ferry functions.

This scheme improves the data delivery performance and also provides a regular connectivity in a disconnected network. The design of a good and optimal ferry route is very important in order to fulfill the bandwidth requirements and to reduce the delays. The problem of ferry route design in case of stationary nodes is NP hard [24] and the details of how the ferry route is designed to solve the delay and buffer requirement problem are given in [24].

In [16] Zhao et al. proposed an efficient message ferry-based approach for data delivery in sparse mobile ad hoc networks. In this work, two variations of MF scheme namely, node-initiated and ferry-initiated MF for message collection have been discussed.

In the *Node-Initiated MF (NIMF)* scheme, ferries move around the deployed area following a predefined known route and communicate with other nodes they meet. Every node in the network knows the path that the ferry follows. The nodes periodically move closer to the ferry and communicate with it. The sender node approaches the ferry and forwards all its messages to the ferry. When the ferry meets the receiver node, it will transfer all these messages related to that particular node. By using the ferry as a relay, the sender node can send messages to the receiver node even though there is no end-to-end path between them. Every node operates in four different modes namely *WORKING, GO TO FERRY, SEND/RECV*, and *GO TO WORK*.

Initially, the node is in the *WORKING* mode and moves around in the network following a random path. The meeting of the nodes with the ferry for sending or receiving the messages is determined by the trajectory control system. The node then proactively moves to meet the ferry for delivering the message. The node enters into the *GO TO FERRY* mode when it decides to visit the ferry. It enters into the *SEND/RECV* mode when it comes in the radio range of the ferry, and then finally exchanges the messages with the ferry. When the message exchange gets completed or the ferry moves out of the radio range, the node enters to the *GO TO WORK* mode. It then returns to its previous location prior to the path taken for message transfer. The node again enters into the *WORKING* mode after returning to its message transfer. The nodes can also enter into the *SEND/RECV* mode from the *WORKING* mode when they accidently meet the ferry without proactive movement i.e., with no intention of message transfer. In this case, after interacting with the ferry, the node again comes back to the *WORKING* mode. The ferry moves according to its fixed route and periodically broadcast a hello message. On reception of the response from a node, it exchanges the message with the node. Node trajectory control mechanism is designed in such a way that the nodes need to keep a balance between performance gain in data delivery and performance degradation in assigned tasks resulting from such proactive movement. The mathematical details of node trajectory control mechanism and message drop rate evaluation in this scheme are given in [16].

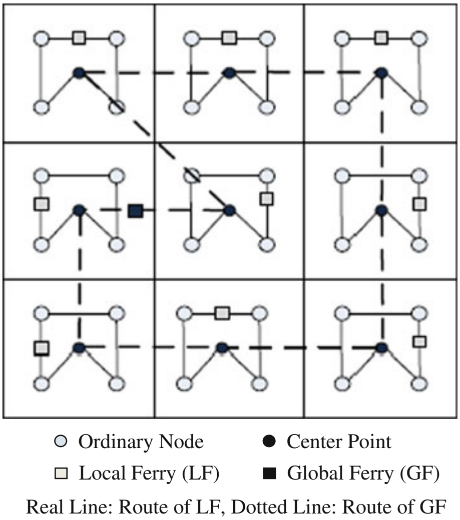
In the *Ferry-Initiated MF (FIMF)* scheme [16], ferries move proactively to meet the nodes in the network. When a node wants to send or receive packets, it generates a *Service Request* and transmits it to a chosen ferry using a long range radio. On reception of a *Service Request*, the ferry will adjust its trajectory to meet with the node and exchange packets using short range radios. To guide the ferry movement, the node occasionally transmits *Location Update* messages to notify the ferry about its new location. In this scheme, a node can operate in two modes namely *DISASSOCIATED* and *ASSOCIATED* and the ferry also operates in two modes namely *IDLE* and *WORKING*.

Initially, a node is in *DISASSOCIATED* mode. After sending a request message to the ferry it enters into the *ASSOCIATED* mode and waits for the interaction with the ferry. In the *ASSOCIATED* mode, the notification control mechanism determines when to send a *Location Update* message to notify the ferry about the node's new location. In both modes, the node may exchange messages with the ferry if it is close to the ferry. When the interaction with the ferry is completed, the node returns back to the *DISASSOCIATED* mode.

The ferry in *IDLE* mode moves on a default route and broadcasts location message periodically. It maintains a set of nodes that have requested the services. Initially this set is empty. On reception of a *Service Request*, the ferry adds that node to the set and enters into the *WORKING* mode. In this mode, it moves according to the computed route which is based on the location of nodes present in the service request node set. It broadcasts the location periodically and on reception of a message from a node the exchange takes place between them. If the message exchange with a certain node is completed, the node is removed from the set of service request nodes. If the set is empty then the ferry enters into the *IDLE* mode else it re-computes the route. The node notification control mechanism details and the performance evaluation of this scheme are given in [16].

In [25] Li et al. proposed a Global Ferry Scheme (GFS) which uses multiple Local Ferries (LF) along with a single Global Ferry (GF) for message transfer in OppNets in order to minimize the average message delivery delay. In GFS, the whole communication area is divided into M square regions. In each region there is

Fig. 5.1 Network model of GFS scheme [25]



a dedicated LF and N/M stationary nodes, where N is total number of nodes in the network. The ordinary nodes can not communicate with each other directly. The LF collects messages from the nodes present in the region and also contacts with the GF to transfer the collected messages when the GF comes in its region. The LF is also responsible for the intra-region communication between the nodes. The global ferry is responsible for facilitating the inter-region communication among the nodes as it can carry and forward messages between different regions. Both LF and GF move with the same speed in the network. The sample network model used in this scheme is shown in Fig. 5.1.

It has been assumed that each ordinary node generates messages of same size with a constant data rate. The transmission time of any message from node to ferry or from ferry to ferry is assumed to be negligible in the calculation of message delivery delay which is the time for the source to the destination delivery. Initially, all LFs are at the center point of their region and GF can start moving from any region. The GF contacts with LF only at the center point of the region. The LF remains at the center point until it meets with the GF for the first time. After meeting with GF, the LF can start moving in its region. Thus, LF moves on a path that passes through every ordinary node in the region and the center point in the region. In this work, through simulations, the performance of GFS is compared with node relaying scheme (NRS) in terms of average delivery delay and delivery ratio in different network configurations. Results show that GFS can improve the performance of OppNets.

In [26] Wang et al. proposed a dynamic message ferry route in partitioned MANETs. In this work, instead of the static nodes mobile nodes with unpredictable traffic in the network have been considered. The nodes in the network are partitioned in various groups called clusters and each group has a node that acts as a cluster head. Each cluster region is called head zone. Ferry nodes are also deployed in the

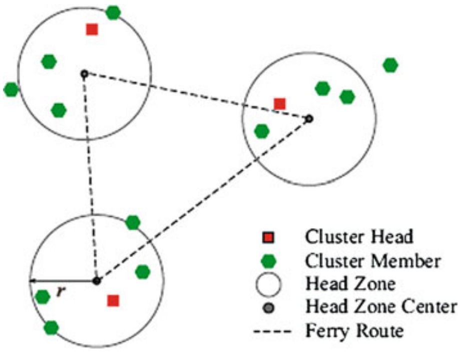


Fig. 5.2 Network topology [26]

network which moves from one head zone to another. Figure 5.2 shows the network topology used in this scheme.

On arriving at a particular cluster, the ferry node will deliver the data to cluster head and decides about its movement to the next head zone. As the members of a particular cluster are connected with each other, the message passing among them can be done using normal MANET routing protocol like ADOV, DSR, etc. The ferry is responsible for the inter-cluster communication.

In this work, the intra-cluster communication time for message passing among the nodes is neglected, since it is very small as compared to the time taken by the ferry to reach from one head zone to another. The weighted average delay of all the messages is given by the equation [26]:

$$\Delta = \frac{\sum_{i=1}^{m} \mu_i \delta_i}{\sum_{i=1}^{m} \mu_i} \tag{5.1}$$

where μ_i is the size of the message i, δ_i, is the time delay that the ferry takes to reach the destination head zone from the source head zone. The main goal of dynamic route design problem is to find the ferry route such that Δ should be minimum. The ferry visits the cluster i either to collect or deliver the messages. It then modifies M_j and W_j for all other j clusters according to the number of undelivered messages present in its buffer. Here M_j is the size of total message for cluster j and W_j is a weight function. For all other clusters, the W_j is computed according to the formula [26]:

$$W_j = \frac{\epsilon_{ij}}{M_j} \tag{5.2}$$

where ϵ_{ij} is the time taken by ferry to move from cluster i to cluster j. If there is no message for cluster j then M_j will be zero, as a result of which W_j will become very large (infinite). The ferry will then move to the cluster whose W_j is minimum. If there are many potential clusters with same minimum W_j then the ferry will move to the least recently visited cluster.

In [27] Zhu et al. described an efficient message ferry routing method for wireless sensor networks. Due to the limited energy and unreliable radio links between the nodes, WSNs are prone to the problem of network partitioning. So the regular methods of routing which do not include message ferrying scheme will fail to deliver the data packets when the source and the destination node reside in two separate partitions. The cost of delivering data from one node to another over a wireless link is directly related to the distance between the two nodes [28]. The energy consumption can be reduced if a message ferry is assigned between the two nodes to relay data. Thus, the MF can be used to save node energy as well as to prolong the network lifetime even though network partitioning does not occur.

The *Tree-based message ferrying* algorithms *(TMFA)* address how the energy is to be saved while keeping delay at a low level for MF routing in WSNs. In these algorithms, minimum-weight spanning trees of each partition of the WSN are evaluated with different alternate source nodes as the root of the spanning trees. Overall energy consumption and delay can be minimized by appropriate choice of the weights for a particular path toward the destination node. If the ferry collects data directly from each sensor node, then the total energy consumption in a WSN is minimum, whereas the delay for all data to reach the sink is maximum [27]. The total energy consumption in the network will increase dramatically if the energy consumption of ferry is also taken into account. In this work, two kinds of tree-constructing algorithms namely *Least Energy Tree (LET)* and *Minimum Hop Tree (MHT)* are discussed and evaluated by deriving an energy model. For comparison of results, *Minimum Spanning Tree (MST)* and *One Level Tree (OLT)* that maximizes the use of the MF are considered. Through mathematical proof, it has been shown that *LET* and *MHT* outperform both *MST* and *OLT* when the message ferry's moving cost is also taken into account [27].

In [29] Mansy et al. proposed a message ferry routing algorithm which is based on *Deficit Round-Robin (DRR)* link scheduling algorithm [30]. It has been observed in [29] that ferry route design problem is similar to the link scheduling problem in traditional networks. Any link scheduling algorithm can be used to solve the route design problem but *DRR* is chosen as it is simple and produces desirable throughput and delay properties in traditional networks. The *DRR* based message ferry routing scheme provides a framework to the ferry route design for stationary nodes in order to optimize the delay performance. A comparison between *Markovian Decision Problem (MDP)* formulation [31] for ferry route and *DRR*-based routing algorithm is drawn. The MDP formulation produces the optimal ferry route that minimizes the average data delivery delay. This scheme is limited to small size networks having nodes with limited size buffers. The nodes with more traffic rate tend to have more visits by the ferry than nodes with relatively lower traffic rate. For message exchange, the nodes relatively closer to each other are visited frequently by the ferry rather than the nodes that are far away from groups of nodes.

The *DRR* algorithm allows the scheduling of multiple flows over a single shared link. The ferry itself represents a resource which is analogous to a link shared among a set of stationary nodes. The nodes are analogous to the flows and the distance between the current position of the ferry and a particular node is similar to the packet size waiting in the flow's queue. A ferry visit to a particular node is analogous to serve a packet from a certain flow. The *DRR* is specifically designed to deal with the flows of unequal and unpredictable packet lengths and does not require the previous knowledge of average packet sizes for the flows. A service quantum counter denoted by *credit[i]* is maintained for each stationary node i. Initially, *credit[i]* $= 0$ for all the nodes. The set of nodes that can be the potential next destination constitutes a list called *candidate_list*. The ferry computes this list in order to get the next node to be visited by it. This computation returns all nodes i whose *credit[i]* is greater than or equal to the distance between the ferry's current location and the node i. In case multiple nodes get selected, a *tiebreak* function [29] is used to select the destination node. The *tiebreak* function filter candidates at various levels like selecting least visited node at first level, nearest node at second level, and so on. The credit value of the selected node is reset to zero. When the *candidate_list* becomes empty, the credit of all the nodes needs to be updated. The updating is done according to their distance from the ferry and their data traffic. Further, optimization can be done by efficiently designing the credit update method and the *tiebreak* function. Simulation results show that the *DRR*-based algorithm produces ferry routes for small problems which are very close to the optimal MDP-based solutions.

In [32] Yang et al. presented the protocols for the ferry replacement in sparse mobile ad hoc networks in which two scenarios for ferry replacement namely the *graceful ferry stop* and the *abrupt ferry failure* are discussed. In *graceful ferry stop*, the ferry informs one or more nodes about its departure before it actually stops functioning as a ferry. In *abrupt ferry failure*, the ferry suddenly crashes due to some unexpected reasons such as system failure or some external attacks such that it does not have the chance to inform other nodes about its departure. Two ferry replacement protocols namely *successor designation protocol* and *distributed election protocol* [32] are discussed in context of the abrupt ferry failure.

In *successor designation protocol*, the ferry chooses its successor among the nodes it meets on its path by seeing their capability above a threshold level. This successor is held responsible for the ferry failure detection and the ferry replacement. A ferry periodically checks the status of the successor during the round trip time and can reassign a new successor if necessary e.g., in case of successor failure. In order to detect the ferry failure, a *status check movement* [32] is used in which the ferry status is checked by the successor regularly after a fixed interval of time. After meeting with the ferry, the successor gets back to its normal operation and sets the timer to a specific value. If the timer expires, the successor moves toward the ferry again. Apart from status checking, the successor may meet the ferry as a regular node for data exchange. Once the data have been exchanged the timer will be reset again. If the ferry either does not find the successor within the expected time or the successor itself informs the ferry to relieve it, then the ferry chooses another successor according to its capability level with respect to a threshold value.

In *distributed election protocol*, nodes elect the ferry among themselves for replacement. Nodes that detect the failure of ferry decide how and when to participate in ferry election. Nodes can participate in the election process as candidate which can become ferry or as an elector that can vote for a candidate to become the ferry. The *backoff* delay [32] scheme is used to reduce the number of unqualified nodes to become a ferry candidate when a ferry failure occurs. Each node's *backoff* time is derived from its capability level, such that the node with the higher capability level will have a shorter *backoff* delay. Each node has to wait for the *backoff* delay duration before it can become a candidate for ferry. If during this waiting time, a candidate is already found then it becomes an elector. The candidates move around the predetermined path and advertise themselves to the electors. They remain around the ferry route till the elections are over. Performance evaluation of the protocols, ferry failure detection, and election are given in [32].

In [33] Zhao et al. presented various algorithms for mobility of the message ferries in DTNs. Employment of multiple ferries increases the cost but at the same time improves the network performance as they can handle more traffic load, can cover greater geographic region and provides reliability in case of a ferry failure.

As there are multiple ferries in the network, so three types of interaction are possible namely *no interaction*, *ferry relaying*, and *node relaying*. In *no interaction*, ferries do not exchange messages while in ferry relaying ferries exchange messages directly and in node relaying the exchange of messages among the ferries take place via a stationary node. Ferry route design with multiple ferries is NP hard and it is designed in such a way to minimize the overall traffic delay in the network [33]. In this work, four algorithms have been proposed for ferry route design which try to minimize the delay. These algorithms have been designed in three phases. First two phases focus on minimizing the total weighted delay and the last phase further extend the ferry routes to meet the desired bandwidth requirements. Different algorithms exploit different type of interaction between the ferries. These algorithms are described below.

(1) *Single-Route Algorithm (SIRA)* In this algorithm, there is a single route on which all the ferries move with same speed and different timings. Thus, a node can communicate with all the ferries in the network. The average delay between two nodes is the sum of the time a node waits for the ferry and the time taken by the ferry to carry that data to the destination [33]. The ferry route is chosen that minimizes the overall weighted delay in the network. To design such a route, *Traveling Salesman Problem (TSP)* [34] is considered. In this case, the optimizing parameter is the overall weighted delay rather than the length of the route. For any particular route, the achieved data rate is directly proportional to the time for which a node remains in contact with the ferry. Therefore, a ferry has to spend more time near the nodes which do not have sufficient time to transmit or receive the messages. This can be achieved by changing the speed of the ferry. Since the speed of the ferry has been assumed to be constant; therefore, there are detours in the ferry route near these nodes and these detours should be as short as possible so as to reduce the delay.

(2) *Multi-Route Algorithm (MURA)* In this algorithm, different ferries follow different paths and data are not relayed between the ferries. The algorithm calculates *estimated weighted delay (EWD)* whenever a node is assigned to a ferry. The EWD is the total delay contributed by the traffic in which source and destination are on the same route and when they are on the different routes. The EWD

depends on the total data rate, weight of the route and maximum data rate of the route [33]. In this algorithm, four types of operations are possible on the ferry routes which are described below.

a. *overlap* It used to overlap the two routes using the overlapping node which incurs minimum EWD. The number of ferries in each route does not change.
b. *merge* It is used to combine two routes into a new route and the total number of ferries in the new route is the sum of the ferries in the individual routes.
c. *merge* It is again used to combine two routes. The only difference is that the number of ferries is one less than the sum of the ferries in both the routes.
d. *reduce* This reduces the number of ferries in a route by one and can only be applied to the routes with more than one ferry.

Initially, every node is assigned to the ferry which incurs minimum EWD. It does so by identifying the best *overlap* or *merge* operation and best *merge-* or *reduce* operation, and then chooses the one with minimum EWD. This process is repeated until the numbers of ferries are reduced to a minimal number. The algorithm further modifies the node assignment to ensure that there is a path between every source and destination pair and the traffic of every route is lower than the maximum capacity of the route.

(3) *Node Relaying Algorithm (NRA)* In this algorithm, ferries interact with each other via a stationary node. The whole network region is divided into cells. Ferries are assigned to the cells that carry data within the cell and also relay the data to the different cells. If the source and the destination are present in different cells, a ferry forwards data to those cells that intersect the line segment joining the source and destination nodes by performing an *overlap* function on the adjacent ferry routes. This algorithm also takes into account the traffic load on each cell e.g., cell present at the center of the grid has large traffic to handle. Initially, a single ferry is assigned to each cell, then the EWD of each route is computed and one of the remaining ferries is assigned to the cell with maximum EWD. The assignment of the ferry to a route decreases the EWD of that route. This assignment continues until all the ferries are allocated to the cells.

(4) *Ferry Relaying Algorithm* This algorithm is similar to NRA with a difference that data are relayed between the ferries directly. Therefore, it accounts for the synchronization of the ferry routes. If the length of the two routes is not same then the ferries meet each other at irregular intervals and they have to carry the message for a longer period of time. It is assumed that the routes have same length and the ferries meet each other at the boundary of the two cells which are called *contact points* [33]. The ferries move in the opposite direction in the two adjacent cells, so that they are synchronized. The length of the ferry route must be proportional to the number of ferries in that route and length of segment between the contact points should be of same length. Initially, a route is constructed that contains only the *contact points*. Further, more nodes are inserted in the route between the contact points that are closest to the node, and this process is repeated till all the nodes are included in the route.

In [35] Ammar et al. discussed about the solution to the *Stability Problem* and *Min-Max Buffer Problem (MMBP)* which arise when message ferries are used in mobile ad hoc networks. The *Stability* problem addresses whether there exist a scheme, such that all the regular nodes have a bounded buffer at any time. The bounded buffer implies that instead of having infinite size buffer every node has a limited amount of buffer size. The *MMBP* describes the design of a ferry route in such a way that minimizes the maximum buffer size required for any node. It is assumed that the node numbered i generates data at the rate r_i and the transfer between ferry and node occurs at the rate r_f. The r_f is assumed to be very high i.e., tending to the infinity. The buffer of every node is bounded if the rate at which a node generates the data are less than the rate at which it is transferred to the ferry between each consecutive visit of the ferry to the same node. It is also assumed that the ferry visits the nodes periodically, because periodic route of the ferry also provides optimal solution to the Min-Max buffer problem [35]. If the data rate of every node is same, then the total buffer filled at any node is directly proportional to the length of the path traveled by the ferry. In other words, the problem of finding the optimal buffer is converted to a *Traveling Salesman Problem* [34]. If the distance between each pair of nodes is unity and the nodes have different data generation rates then there exists 4/3 factor approximation algorithm [35] for *Min-Max buffer problem*. This is calculated by modeling the problem into a *pin-wheel scheduling problem* [36–41]. From the *pin-wheel scheduling* algorithm it is calculated that

$$\text{OPT} \geq \sum_{n=1}^{K} (r_n) \tag{5.3}$$

where r_n is the rate of data generation by the node n and OPT is the optimal value of buffer size.

5.2.3 Infrastructure in Form of Access Points and Data MULEs

In [17] Jain et al. described an energy efficient data retrieval method by exploiting the mobility of some special nodes called Mobile Ubiquitous LAN Extensions (MULEs) in wireless sensor networks. In WSNs, the nodes not only have to transfer their own data but also forward the data of some other nodes which result in lot of energy consumption. So a method is needed to minimize the energy loss of the sensor nodes in order to maximize their life. This can be achieved by employing the MULEs in the network that passes through the sensor nodes. These MULEs have large buffer, capable of short-range wireless communication and receive data from the sensor nodes and deliver them to the access points. Figure 5.3 shows the architecture of the MULEs system.

The MULE architecture has been divided into three tiers. The lower tier is composed of sensor nodes that periodically collect the data from the surroundings. The

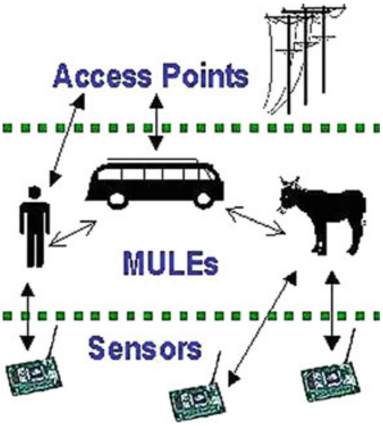


Fig. 5.3 The three tier system [17]

middle tier consists of mobile nodes named MULEs which move around the sensor networks to gather the data collected by the sensor nodes. The upper tier has fixed access points which receive the data gathered by the MULEs in the middle tier. These access points are responsible for processing and storage of the information received from the MULEs. The MULE architecture is energy efficient, scalable, and reliable through acknowledgments. But it suffers from data latency which can be removed by proper positioning of the sensor nodes, access points, the MULEs mobility model, and many other parameters.

The MULE system is designed to reduce data success rate, latency, and energy spent in communication between the MULE and the sensor node. Success rate is the ratio of the data transferred to the access points to the data generated and latency is the total time spent from the generation of the message to its delivery to the access point. An analytical model has been proposed in [17], which tries to minimize the above-mentioned parameters. Energy consumption of the sensor nodes can be minimized by reducing the sensor duty cycle, which is done by increasing the contact time between MULE and the sensor node [17]. Latency can be further reduced by having MULE-to-MULE communication. Simulation results under various parametric studies can be referred in [17].

5.2.4 Infrastructure in Form of Buses and Road Side Units

In [42] Wang et al. described a Traffic Infrastructure-Based Cluster Routing Protocol with Handoff *(TIBCRPH)* in *Vehicular ad hoc networks (VANETs)*. In this scheme, it is assumed that each node is aware of its own position through *Global Positioning System (GPS)*. The source node also knows the location and velocity of the destination node. Due to high mobility, the nodes in VANET can suffer frequent breaks in their routes. The existing traffic infrastructure is used to divide the whole network area into multiple clusters in order to achieve the efficient transmission of the messages. Each cluster has a cluster head and some member nodes associated with it. The existing traffic infrastructures are used as cluster heads within the cluster. The members of a cluster can communicate with each other directly and the cluster heads can communicate either through wired or wireless mode of communication. The clusters may overlap with each other and a *handoff metric* [42] is needed to find the new cluster head of the vehicles when they move across the overlapped region. The dot product of a vehicle's velocity vector and the direction vector of its two neighboring cluster heads is calculated. The product is used to determine the ID of the most suitable cluster head among the two for this vehicle. This selected node will then be the new cluster head of the vehicle and the vehicle will communicate with other vehicles through this new cluster head.

In order to send a message using this protocol, the source node first obtains the location of itself and the destination node through *GPS*. If the source node is in one hop range of the destination node, it sends the message directly to the destination node. Otherwise, the source node puts the location of the destination node in the message header. The source then finds its cluster head (say M1) using the *handoff metric* and passes the message to M1. Using the *handoff metric*, M1 finds the cluster head (say M2) of the destination node and sends the message to M2 via the backbone network. M2 then extracts the destination location from the message and finally passes the message to the destination node.

In [43] Luo et al. proposed a Mobile Infrastructure-Based VANET Routing protocol *(MIBR)* in urban environment. Two different types of vehicles namely buses and cars are employed which constitute the road traffic. The buses are used as the mobile infrastructures which participate in the routing and forwarding of packets. They constitute the mobile backbone to facilitate multi-hop communication between the vehicles. The buses can improve the network connectivity as they have fixed travel lines and powerful sending abilities with a larger transmission range. Each bus has two wireless interfaces which work on two different channels, while the ordinary cars have only a single interface. The first channel is used for communication between two cars or between a car and a bus with a transmission range $R1$. The second channel is used for communication between the buses with a transmission range $R2$ such that $R2 > R1$.

It has been assumed in [43] that vehicles move in form of clusters due to the traffic lights. Each vehicle knows its location through *GPS* and has a digital street map with it. The source vehicle can also get the destination information with the help of a *location service* mechanism. Each vehicle periodically sends beacon packets which contain its location information and the information about the type of the vehicle (Bus or Car). The working of the *MIBR* protocol is divided in two parts:

1. *Selection of an optimal route* In *MIBR*, the route consists of a sequence of road segments which are chosen one by one with the best transmission quality. The number of buses present in a road segment is proportional to the density of the total vehicles moving on the road i.e., more is the number of buses, more is the number of vehicles on the road segment. The hop count of each road segment is calculated

by the expected density of the buses present on the segment and the length of the road. The road segment and its estimated hop count are stored in a route table. The mathematical details of the hop count calculation for each segment can be referred from [43]. The sending vehicle or the intermediate vehicle near the junction selects the next forwarding road segment by checking the route table. It chooses the best neighboring road segment with minimum estimated hop count to the destination. Once the expected hop count for each road segment is known, the total hop count for a particular path can be easily calculated.

2. *Efficient forwarding of the packets* The *MIBR* protocol uses the *"bus first"* strategy to forward the packets on the road segments. The *MIBR* prefers buses as the next hop instead of the ordinary vehicles due to their longer transmission range. When the packet approaches a junction, the vehicles on the next road segment have higher priority than the vehicles on the same road to be selected as the next hop for packet forwarding. The *"bus first"* strategy used to select the forwarding next hop vehicle is summarized below [43].

    - If the next road segment contains any buses, then the bus closest to the junction after the next junction is chosen. Otherwise, the ordinary car which is closest to the junction after the next junction is chosen.
    - If there are no vehicles on the next road segment and the packet is being carried by a bus, then a bus is chosen on the same segment which is closest to the next junction. Otherwise, a vehicle is chosen on the same segment which is closest to the next junction.
    - If there are no vehicles on the next road segment and the packet is currently being carried by a car, then a bus is chosen on the same segment which is closest to the next junction. Otherwise, a vehicle is chosen on the same segment which is closest to the next junction.
    - If no better forwarding vehicles are available then the packet is dropped.

5.3 Conclusion

In this chapter, a brief outline of various routing protocols for the infrastructure-based OppNets have been presented. These protocols use some form of infrastructure for routing and forwarding of messages in this type of network. The infrastructure used by them is either fixed infrastructure which is present in form of infostations or mobile infrastructure which is present in form of message ferry and data MULEs. The message ferrying scheme is discussed under various network environments like wireless sensor networks, highly partitioned and sparse mobile ad hoc network, etc. These schemes provide various types of benefits in different network scenarios. For example, tree-based message ferrying scheme saves the energy of regular nodes and other message ferrying schemes discussed reduces the delay and the buffer space requirement in the message delivery. Various ferry replacement techniques have also been discussed to handle the case of ferry failure. While this chapter has not covered all the routing protocols available in infrastructure-based OppNets, it tries to give a summary of the research techniques used by discussing a few protocols designed in this area. Further, this study even identifies some important and explicit characteristics of each protocol along with their areas of application.

References

1. Lilien L, Kamal ZH, Bhuse V, Gupta A (2006) Opportunistic networks: the concept and research challenges in privacy and security. In: Proceedings of NSF international workshop on research challenges in security and privacy for mobile and wireless networks (WSPWN 2006), Miami, March 2006, pp 134–147
2. Chen L-J, Hung Yu C, Tseng C, Chu H, Chou C (2008) A content-centric framework for effective data dissemination in opportunistic networks. IEEE J Sel Areas Commun 26(5):761–772
3. Perkins CE, Royer EM (1999) Ad hoc on-demand distance vector routing. In: Proceedings of the 2nd IEEE workshop on mobile computing systems and applications, New Orleans, LA, pp 90–100
4. Dhurandher SK, Sharma DK, Woungang I, Chao HC (2011) Performance evaluation of various routing protocols in opportunistic networks. In: Proceedings of IEEE GLOBECOM workshop 2011, Houston, Texas, 5–9 Dec 2011, pp 1067–1071
5. Huang C-M, Lan K-C, Tsai C-Z (2008) A survey of opportunistic networks. In: Proceedings of the 22nd international conference on advanced information networking and applications—workshops (AINAW 2008), Okinawa, Japan, 25–28 March 2008, pp 1672–1677
6. Zhang Z (2006) Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges. IEEE Commun Surv Tutor 8(1):24–37
7. Fall K (2003) A delay-tolerant network architecture for challenged internets. In: Proceedings of ACM SIGCOMM 2003, Karlsruhe, Germany, 25–29 Aug 2003, pp 27–34
8. Jain S, Fall K, Patra R (2004) Routing in a delay tolerant network. In: Proceedings of ACM SIGCOMM 2004, 30 Aug–3 Sept 2004, pp 145–158
9. Pelusi L, Passarella A, Conti M (Nov 2006) Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. IEEE Commun Mag 44(11):134–141
10. Vahdat A, Becker D (2000) Epidemic routing for partially connected ad hoc networks, Technical report CS-2000-06. Duke University, Department of Computer Science, Durham
11. Spyropoulos T, Psounis K, Raghavendra CS (2005) Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In: Proceedings of ACM SIGCOMM workshop on delay-tolerant networking (WDTN 05), Philadelphia, PA, 22–26 Aug 2005, pp 252–259
12. Spyropoulos T, Psounis K, Raghavendra CS (2007) Spray and focus: efficient mobility-assisted routing for heterogeneous and correlated mobility. In: Proceedings of the fifth IEEE international conference on pervasive computing and communications workshops (PerComW '07), White Plains, 19–23 March 2007, pp 79–85
13. Lindgren A, Doria A, Schelen O (July 2003) Probabilistic routing in intermittently connected networks. ACM SIGMOBILE Mob Comput Commun Rev 7(3):19–20
14. Boldrini C, Conti M, Iacopini I, Passarella A (2007) HiBOp: a history based routing protocol for opportunistic networks. In: Proceedings of IEEE international symposium on world of wireless, mobile and multimedia networks (WoWMoM 2007), Espoo, Finland, 18–21 June 2007, pp 1–12
15. Goodman DJ, Borras J, Mandayam NB, Yates RD (1997) INFOSTATIONS: a new system model for data and messaging services. In: Proceedings of IEEE vehicular technology conference 1997(VTC97), vol 2, May 1997, pp 969–973

16. Zhao W, Ammar M, Zegura E (2004) A message ferrying approach for data delivery in sparse mobile ad hoc networks. In: Proceedings of 5th ACM international symposium on mobile ad hoc networking and computing 2004 (MobiHoc'04), ACM Press, Tokyo, 24–26 May 2004, pp 187–198
17. Jain S, Shah RC, Bbrunette W, Borriello G, Roy S (June 2006) Exploiting mobility for energy efficient data collection in wireless sensor networks. ACM/Kluwer Mob Netw Appl (MONET) 11(3):327–339
18. Cavalcanti D, Sadok D, Kelner J (2002) Mobile infostations: a paradigm for wireless data communications. In: Proceedings of IASTED international conference on wireless and optical communications (WOC 2002), 17–19 July 2002, pp 441–446
19. Small T, Haas ZJ (2003) The shared wireless infostation model—a new ad hoc nNetworking paradigm (or where there is a whale, there is a way). In: Proceedings of 4th ACM inernational symposium on mobile ad hoc networking and computing (MobiHoc 2003), Annapolis, MD, 1–3 June 2003, pp 233–244
20. Brauer F, Castillo-Ch'avez C (2001) Mathematical models in population biology and epidemiology. Springer, New York, ISBN 978-1-4419-3182-5
21. Mate BR, Gisiner R, Mobley J (1998) Local and migratory movements of Hawaiian humpback whales tracked by satellite telemetry. Can J Zool 76(5):863–868
22. Krutzikowsky GK, Mate BR (2000) Dive and surfacing characteristics of bowhead whales (Balaena mysticetus) in the Beaufort and Chukchi seas. Can J Zool 78:1182–1198
23. Mate BR, Lagerquist BA, Calambokidis J (1999) The movements of North Pacific blue whales during the feeding season off Southern California and their southern fall migration. Mar Mammal Sci 15(4):1246–1257
24. Zhao W, Ammar M (2003) Message ferrying: proactive routing in highly-partitioned wireless ad hoc networks. In: Proceedings of the ninth IEEE workshop on future trends of distributed computing systems (FTDCS). San Juan, Puerto Rico, 28–30 May 2003, pp 308–314
25. Li Y, Weng B, Liu Q, Lijun Tang, Daneshmand M (2011) Multiple ferry routing for the opportunistic networks. In: Proceedings of IEEE Globecom 2011, Houston, Texas, 5–9 Dec 2011, pp 1–5
26. Wang T, Low CP (2010) Dynamic message ferry route (dMFR) for partitioned MANETs. In: Proceedings of international conference on communications and mobile computing, Shenzhen, 12–14 April 2010, pp 447–451
27. Zhu Y-H, Wu W-D, Leung VCM, Yang L-H (2008) Energy-efficient tree-based message ferrying routing schemes for wireless sensor networks. In: Proceedings of third international conference on communications and networking in China (ChinaCom), Hangzhou, 25–27 Aug 2008, pp 844–848
28. Raghunathan V, Schurgers C, Sung P (2002) Energy-aware wireless microsensor networks. IEEE Signal Process Mag 1(7):40–50
29. Mansy A, Ammar M, Zegura E (2011) Deficit round-robin based message ferry routing. In: Proceedings of IEEE GLOBECOM workshop 2011, Houston, Texas, 5–9 Dec 2011, pp 1–5
30. Shreedhar M, Varghese G (1995) Efficient fair queueing using deficit round-robin. Proc SIGCOMM 25(4):231–242
31. Mansy A, Ammar M, Zegura E (2011) Deficit round-robin based message ferry routing. Georgia Tech, Atlanta, Technical report GT-CS-11-03
32. Yang J, Chen Y, Ammar M, Lee C (2005) Ferry replacement protocols in sparse MANET message ferrying systems. In: Proceedings of IEEE wireless communications and networking conference, New Orleans, 13–17 March 2005, pp 2038–2044
33. Zhao W, Ammar M, Zegura E (2005) Controlling the mobility of multiple data transport ferries in a delay-tolerant network. In: Proceedings of the 24th annual joint conference of the IEEE computer and communications societies (INFOCOM05) 2005, Miami, vol 2, pp 1407–1418
34. Papadimitriou CH, Yannakakis M (1993) The traveling salesman problem with distances one and two. Math Oper Res 18(1):1–11
35. Ammar M, Chakrabarty D, Sarma AD, Kalyanasundaram S, Lipton RJ (2009) Algorithms for message ferrying on mobile ad hoc networks. In: Proceedings of the IARCS annual conference

on foundations of software technology and theoretical computer science (FSTTCS09), IIT Kanpur, India, 15–17 Dec 2009, pp 13–24

36. Chan MY, Chin Francis YL (1992) General schedulers for the pinwheel problem based on double-integer reduction. IEEE Trans Comput 41(6):755–768
37. Chan MY, Chin FYL (1993) Schedulers for larger class of pinwheel instances. Algorithmica 9(5,):425–462
38. Fishburn PC, Lagarias JC (2002) Pinwheel scheduling: achievable ensities. Algorithmica 34(1):14–38
39. Holte R, Mok A, Rosier L, Tulchinsky I, Varvel D (1989) The pinwheel: a real-time scheduling problem. In: Proceedings of the 22nd Hawaii international conference of system sciences, Jan 1989, pp 693–702
40. Holte R, Rosier L, Tulchinsky I, Varvel D (1992) Pinwheel scheduling with two distinct numbers. Theor. Comput. Sci. 100(1):105–135 (Mathematical foundations of computer science 1989 (Porpolhk abka-Kozubnik 1989), Lecture notes in computer science, vol 379, 1989, pp 281–290)
41. Lin S-S, Lin K-J (1997) A pinwheel scheduler for three distinct numbers with a tight schedulability bound. Algorithmica 19(4):411–426
42. Wang T, Wang G (2010) TIBCRPH: traffic infrastructure based cluster routing protocol with handoff in VANET. In: Proceedings of 19th IEEE annual wireless and optical communications conference (WOCC), Shanghai, 14–15 May 2010, pp 1–5
43. Luo J, Xinxing G, Zhao T, Yan W (2010) A mobile infrastructure based VANET routing protocol in the Urban environment. In: Proceedings of IEEE international conference on communications and mobile computing (CMC 2010), Shenzhen, 12–14 April 2010, pp 432–437

Chapter 6
Opportunistic Routing in Mobile Ad Hoc Networks

Pramita Mitra and Christian Poellabauer

Abstract Routing in opportunistic mobile ad hoc networks (MANETs) is characterized by intermittent and sporadic communication opportunities between portable devices. As a result, a path between the source and the destination(s) may only exist for a brief and unpredictable period of time, thereby leading to network partition. Most existing routing approaches store data packets locally and forward them when in reach of a mobile peer that carries the data packets closer to the destination. Through a survey of recent literature, this chapter will provide an overview of the routing protocols that exploit intermittent communication opportunities for increased data transmission success in MANETs, and characterize them using the following three aspects: (1) how does the protocol discover and exploit the intermittent communication opportunities in a mobile environment? (2) How does the protocol address multicast, which is a common data dissemination pattern in an opportunistic communication environment? and (3) What kind of real-world applications are enabled by the use of opportunistic routing protocols?

6.1 Introduction

Current trends in networking-architecture developments, e.g., delay and disruption-tolerant networks, aim to deal with the disconnections that naturally and frequently arise in wireless environments. There are many asynchronous applications, e.g., e-mail, multimedia messaging, cached web access, file sharing, and blogging, which do not require continuous network access. *Opportunistic networks* (OPNETs) is a fast emerging paradigm that is created out of mobile devices carried by people, without relying on any preexisting network topology [1]. In OPNETs, network irregularities

---

P. Mitra (✉) · C. Poellabauer
Department of Computer Science and Engineering, University of Notre Dame,
Notre Dame, IN, USA
e-mail: pmitra@nd.edu

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_6,

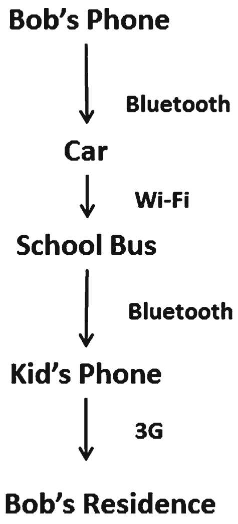


Fig. 6.1 An example of exploiting opportunistic communications

such as frequent disconnection, network partition, and node mobility are considered as norms. To that end, some of these irregularities are even used as techniques to provide communication opportunities between disconnected groups of nodes, rather than a drawback to be solved.

Because of the above unconventional approach to handle network irregularities, routing is one of the most challenging issues in the design of OPNETs. To that end, a complete path between source and destination is often unavailable. OPNET routing protocols solve this problem by removing the assumption of physical end-to-end connectivity and allow disconnected nodes to exchange messages. In OPNETs, intermediate nodes store messages when there is no forwarding opportunity toward the destination, and exploit any future contact opportunity with other mobile devices to bring the messages closer to the destination [2–4]. Figure 6.1 illustrates an example of how opportunistic communications can be exploited to exchange messages between disconnected nodes in an urban environment. As shown in Fig. 6.1, Bob wishes to send a message to his family but has no 2G/3G connectivity. He opportunistically forwards the message using the Bluetooth radio in his phone to another car driving in the same direction as his residence. The car moves through the traffic in the city, and transfers the message using a Wi-Fi link to a school bus going in the same direction as Bob's residence. Next, the bus forwards the message using its Bluetooth radio to the cellphone of a kid who is disembarking at bus stop in the same area where Bob lives. When the kid walks past Bob's residence on his way to home, the message is finally delivered to its destination. As is shown in this example, a network connection between the source and destination was never available; however, by opportunistically exploiting contacts among heterogeneous devices, the message is delivered hop-by-hop closer to the destination, and eventually is received by the destination.

Through a survey of the key research approaches to routing in OPNETs, this chapter provides a detailed overview of the routing protocols that explicitly focus on exploiting connection opportunities for routing in an intermittently connected network. The remainder of this chapter is organized as follows: Section 6.2 discusses the various aspects of routing in OPNETs and proposes an analytical framework to characterize the recent relevant research approaches. Section 6.3 provides a detailed overview of a number of routing protocols designed for OPNETs. Section 6.4 discusses a few applications that are enabled by use of opportunistic communications. Section 6.5 identifies a few challenges in current research efforts and Sect. 6.6 concludes the chapter.

6.2 What is an Opportunistic Mobile Ad Hoc Network?

Opportunistic Networks is an emerging communications paradigm that supports a "whenever, wherever" style of communications between mobile nodes. In OPNETs, no assumption is made about the availability of an end-to-end path between the source and destination(s)—in fact they might never be connected to the same network at the same time, but still be able to exchange messages between them.

Figure 6.1 illustrates an example of how communication opportunities change with time in OPNETs due to node mobility. OPNETs typically comprise heterogeneous wireless devices in motion (e.g., portable devices carried by mobile users, vehicles, etc.) that form dynamic clusters or islands of connectivity. These devices take advantages of radio contacts with peers as and when they become available, and cooperate in routing data. It may take multiple discontinuous wireless contacts or hops before the data is transferred from the source to the destination over a period of time. As shown in Fig. 6.1, there is no direct path from node S to node D at any given time. Packets from S can be delivered to D if intermediate nodes store-and-forward the packets in the following fashion: at time t_1, node S sends the packet to node 2; at $t_2 > t_1$, node 2 forwards the packet to node 3; and at $t_3 > t_2 > t_1$, node 3 forwards the packet to node D.

This section presents a set of definitions of OPNETs, elaborated by recent related research efforts, which in turn provide guidelines for building a framework that is used to analyze these existing research efforts.

6.2.1 Definition

The term "opportunistic" with regard to wireless networking implies the tendency of network devices to exploit available resources in the network as and when possible. Several concepts behind OPNETs originate from the research efforts on *delay-tolerant networks* (DTN) that has been conducted within the Internet Research Task

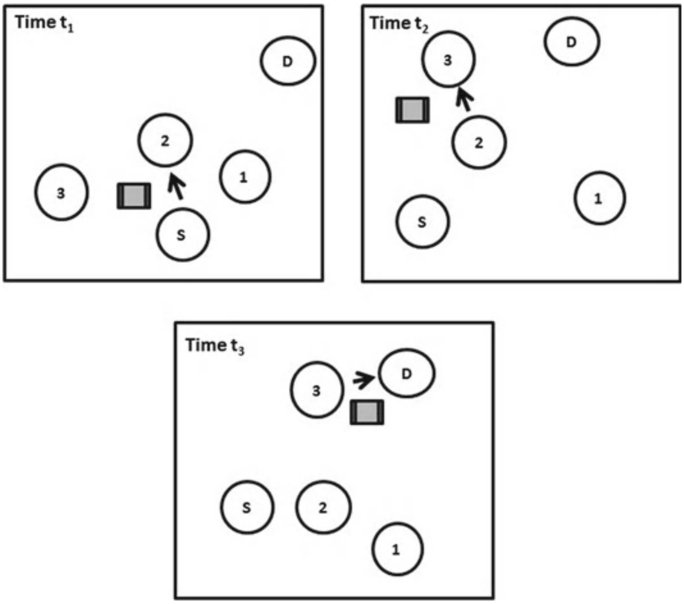


Fig. 6.2 An example of how node mobility creates communication opportunities in OPNETs at time interval t_1, t_2 and t_3 ($t_3 > t_2 > t_1$)

Force.[1] DTNs are characterized by their lack of connectivity, resulting in unavailability of instantaneous end-to-end paths between source and destination(s). As a result, DTN routing protocols must take to a *"store-and-forward"* approach, where data is incrementally moved and stored throughout the network in hope that it will eventually reach its destination(s). Pelusi et al. [3] argues that there is no clear separation of concepts for OPNETs and DTNs; however, OPNETs correspond to a more general and flexible concept which includes DTNs. To that end, DTNs assume a priori knowledge of Internet-like topologies in which several links between gateways could be available just at certain (possibly unspecified) times, and routes are typically computed via legacy Internet techniques by taking into consideration the link unavailability just as another component of link cost. On the other hand, in OPNETs it is not mandatory to have a priori knowledge of the network topology—instead, routes are computed at every hop before a packet is forwarded and thus each single node acts as a gateway. Phanse et al. [5] proposed a more formal definition of OPNETs, which is the following: "Opportunistic networks are intrinsically *fault tolerant* for they are not limited by the end-to-end connectivity assumption. These networks are *distributed* and *self-organizing* in that the control and management is largely up to the individual devices or users (within the boundaries defined by the network operator's policies, if part of a commercial network). The communication in these networks is *localized*, i.e., decisions such as routing are made by devices

---

[1] http://www.dtnrg.org/docs/specs

based on locally available information. Opportunistic also means being able to take advantage of *locally accessed global information*, where devices implicitly convey global reachability information strictly through local interaction."

In the last few years, there have been a lot of research activities focusing on various applications of multi-hop *mobile ad hoc networks* (MANETs). Originally conceived for military applications, MANETs have recently found increasing applications in many civil scenarios such as search and rescue operations, sensor networks, connecting students on campus, sharing free Internet connection etc. MANETs are similar to OPNETs in that neither network relies on a centralized entity; their architecture is decentralized by definition. Due to node mobility, nodes connect and disconnect as they move in and out of communications range. Connection and disconnection may also happen because devices are turned on or off unpredictably. However, MANET routing algorithms focus on end-to-end communications paradigm, which takes both user mobility and limited resources into account. Link failures due to frequent connection/disconnection are treated as anomaly and handled by employing route management/recovery strategies. On the other hand, routes are formed online in OPNETs with the data message being forwarded one link at a time, as links in the route become available. OPNETs are, therefore, an extension of MANET with the end-to-end communications paradigm relaxed or removed, if necessary. Hence, the rest of this chapter will refer to OPNETs as *opportunistic mobile ad-hoc networks* (OPMANETs).

6.2.2 Challenges of Routing in OPMANETs

OPMANETs are highly challenged networks with intermittent connectivity and variable link performance. In such environments, complete end-to-end routes from the source node to the destination(s) may not exist from time-to-time or at all. In addition, the links are unstable and may change or break quickly. In order to make communications possible in OPMANETs, the routing protocols often extend the "store-and-forward" approach used in DTNs to "store-carry-forward" approach—in this approach, there are situations when a suitable next-hop node may not be immediately found for the current node to forward the data to. In that case, the current node will need to buffer the data until the node gets an opportunity to forward the data and must be capable of buffering the data for a considerable duration [2]. The difficulties in designing a protocol for efficiently and successfully routing messages to their destination(s) in OPMANETs are outlined in Table 6.1.

6.2.3 Analysis Framework

Using these above perspectives, this book chapter proposes a simple framework to analyze the existing routing protocols in OPMANETs, choosing three main axes to characterize them. These axes refer to the following questions:

Table 6.1 Challenges in routing in OPMANETs

Designing a routing protocol for OPMANETs requires many challenges to be addressed. For example:

- *Bad Forwarding Decision*. Due to intermittent connectivity and node mobility in OPMANETs, nodes connect and disconnect to and from the network frequently. As a result, nodes often make bad forwarding decisions, i.e., they fail to forward data to the best relay node(s)
- *Connection Unpredictability*. Making routing decisions in OPMANETs is a non-trivial problem. To that end, it is highly difficult to determine which nodes are the best carriers because the contact duration and data transfer rate are highly unpredictable and intermittent
- *Data Dissemination*. The main idea for OPMANET applications is to share physical resources and sensor information with other devices in the network in a spontaneous and ad hoc fashion. It is not easy to provide reliable multi-hop, multi-destination communications in OPMANETs with intermittent and unpredictable connectivity
- *Resource Constraints*. The devices in OPMANETs are handheld mobile devices with limited battery power, computational capabilities, and storage. As OPMANETs could be very sparse, the current node may have to buffer data for a long period of time and also from time-to-time keep trying to find a suitable node to forward the data to—thereby leading to drain on the local resources. This situation can become worse when a bad forwarding decision is made, which may cause the packets to be delayed indefinitely
- *Privacy and Security*. In order to support data or communications authenticity and integrity and provide protection of user identity, strong security and privacy algorithms need to be implemented. However, the available algorithms are too complex or run too slow on resource-constrained mobile devices
- *User Interaction*. The mobile devices in OPMANETs are mostly personal devices carried by human users. Most applications in OPMANETs are pervasive in the sense that they would not require active user participation for most of time; however, some of the received data may require some kind of instantaneous reaction. It is not easy to design a general user interaction model for OPMANETs because the user interaction requirements vary depending on the application, urgency of the delivered information, and device capabilities

- *Discovery and Dissemination*. How does the protocol discover and exploit the intermittent communication opportunities in a mobile environment? How does the protocol provide reliable multi-hop, multi-destination data delivery, which is a common data dissemination pattern in an opportunistic communication environment?
- *Resource Usage*. How does the protocol impact on the battery power, CPU usage, and memory of the local device while participating in a peer-to-peer opportunistic communication environment with intermittent connectivity?
- *Applications*. What kind of real-world applications are enabled by the use of OPMANET routing protocols? What are their quality of service (QoS) requirements?

The remainder of this chapter will use these aspects to characterize a number of recent routing protocols in OPMANETs.

6.3 Routing in Opportunistic Mobile Ad Hoc Networks

Routing is the one of the most compelling challenges in the design of OPMANETs due to intermittent and unpredictable connectivity, lack of knowledge about the frequently changing network topology, and resource constraints of mobile devices. Figure 6.3 shows a possible taxonomy of the routing algorithms in OPMANETs with respect to how they disseminate data to destinations. Based on the number of copies of a data message forwarded by each node, the routing algorithms can be broadly classified into four categories: *Forwarding-based* (*single copy*) *approach, Flooding-based* (*multiple copies*) *approach*, *Hybrid approach*, and *Social behavior-based approach*. In the forwarding-based approach, the data message is forwarded to only one single node in each step of routing. This approach tries to reduce the buffer usages and the number of messages transferred in the network at the expense of increased CPU usage (for finding a suitable next-hop node), long delays, and low delivery ratios. On the other hand, flooding-based approaches may generate multiple copies of the same data message—the copies of the message can either be spread in the network like a disease epidemic or each copy can be routed separately for increased efficiency and robustness. This approach achieves lower delays and higher delivery ratio at the cost of increased storage requirement and higher number of data message transmissions. The hybrid approaches combine features from forwarding-based approaches with flooding-based approaches to achieve low data dissemination latency while limiting the network overheads. Lastly, the social behavior-based approaches make forwarding decisions using locally observed patterns about social ties between nodes to predict future contact opportunities. This

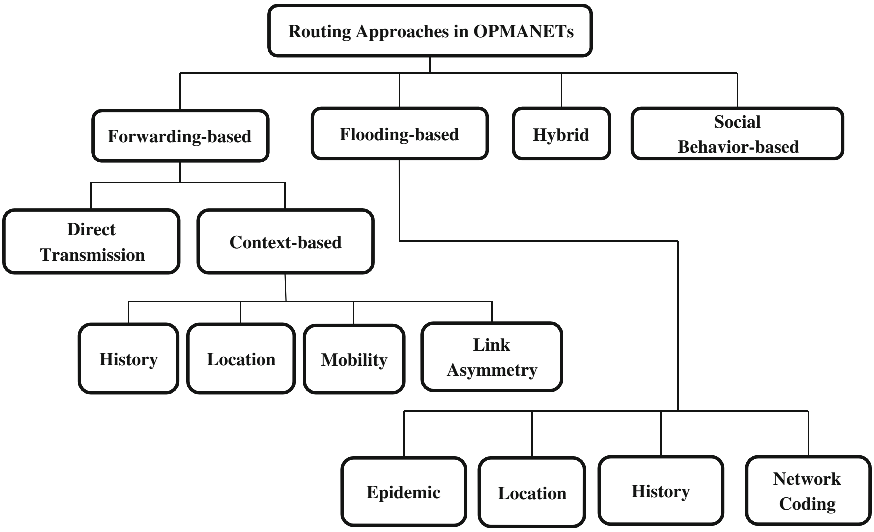


Fig. 6.3 A taxonomy of routing approaches in OPMANETs

section provides an overview of these various routing algorithms and characterizes them using the analytical framework proposed in Sect. 6.2.

6.3.1 Forwarding-Based Approaches

Based on how the forwarding decision is made (i.e., how the best next-hop node is chosen), the forwarding-based routing approaches can be divided into the following two sub-categories, (1) *direct transmission-based forwarding*—protocols in this category limit the number of hops traveled by a data message, and (2) *context-based forwarding*—protocols in this category exploit the context in which the nodes are operating to decide which node to forward the data message to as well as whether it should transmit the message immediately or hold the message until it meets a better node. Context-based forwarding can further be divided into four sub-sub-categories, i.e., (1) *history/estimation-based forwarding*, (2) *location-based forwarding*, (3) *mobility-based forwarding* and (4) *link asymmetry-based forwarding*. History/estimation-based protocols allow a node to make better forwarding decision by estimating the forwarding probability of its neighbors based on historical data. Location-based forwarding protocols use locations of nodes as the context to make forwarding decisions—the best next-hop node is the one that moves the data message closest to the destination. Mobility-based protocols allow a node to make better forwarding decision by estimating the forwarding probability of its neighbors based on mobility patterns/traces. Link asymmetry-based protocols exploit asymmetric communication opportunities in heterogeneous OPMANETs for increased data transmission success and lower latency. This subsection provides a detailed overview of the forwarding-based approaches.

6.3.1.1 Direct Transmission-Based Forwarding

Direct transmission-based forwarding approaches minimize the number of data message transmissions by limiting the number of hops traveled by the data message between source and destination(s). Spyropoulos et al. [6] proposed a 1-hop forwarding approach which allows the source to hold onto the data message and deliver to the destination(s) only when they are within transmission range. This approach has minimal overhead, but the delay could be very long as it is unbounded. Grossglauser et al. [7] proposed a 2-hop forwarding approach based on a theoretical framework, in which nodes freely roam around the network and every node gets close to any other node for a random time duration per time slot. Within this framework, a node s gives a message, addressed to destination node t, to another randomly chosen 1-hop neighbor r. The node r transmits the message to the destination t when it happens to be within the communications range of t. Hence, a message will only travel 2-hops and no message will be transmitted more than twice. This scheme claims that a message

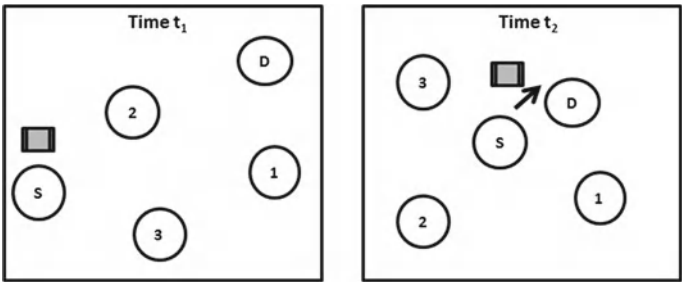


Fig. 6.4 Example of 1-hop direct transmission-based forwarding approach [6]

is always guaranteed to be delivered, even if its delivery time is averaged over many time slots.

6.3.1.2 Context-Based Forwarding

Context-based forwarding approaches exploit the context in which nodes are operating so as to identify suitable next hops toward the eventual destinations. Context-based forwarding approaches can be divided into the following four types:

1. *History/estimation-based forwarding*. History/estimation-based protocols allow a node to make better forwarding decision by estimating the forwarding probability of its neighbors based on historical data. Protocols in this category differ in the parameters they use for making this estimation; commonly used parameters are previous connection/disconnection time between nodes, local residual battery level, mobility rate, and number of messages forwarded by a neighbor in the past, etc. Jain et al. [8] proposed a knowledge-based routing scheme, which is the first study in this area. Depending on the amount of knowledge available about network topology characteristics and traffic demand, they defined four knowledge oracles. Each oracle presents some particular knowledge of the network, e.g., schedule of meetings between nodes. This approach applies traditional shortest path routing techniques to OPNETs by exploiting the knowledge oracles. Messages are forwarded over the shortest path using source routing techniques. This approach was extended by Jones et al. [9], who proposed *minimal estimated expected delay* (MEED) routing. MEED computes the expected delay using the observed contact history, in which a node records the connection and disconnection time of each contact over a sliding history window. Link-state updates are propagated to all nodes in the network using an epidemic link-state protocol. The routing table is recomputed each time a contact arrives and before a message is to be forwarded, resulting in per contact routing. Tan et al. [10] proposed *a shortest expected path routing* (SEPR) to maintain a topology map at each node to every other node. SEPR calculates the link forwarding probability based on history data of the duration of last contact between two nodes. Each message stored in buffer is assigned an effective path length (EPL); the value is set at infinity when it is

inserted in the cache for the first time. When two nodes meet, they exchange the EPL values—a smaller EPL value suggests a higher probability of delivery. When a node receives a smaller EPL, it will update its local EPL value. As a result, during the cache replacement procedure, those messages with smaller EPL values to the destination are removed first. EPL is also used in deciding which nodes to forward the messages. Using SEPR protocol, the same message could be forwarded to multiple nodes to increase reliability and to reduce delay. Musolesi et al. [11] proposed *context-aware routing* (CAR) protocol in which each node in the network is in charge of producing its own delivery probabilities toward each known destination node. Delivery probabilities are exchanged periodically so that, eventually, each node can compute the best carrier for each destination node. The best carriers are computed based on the multiattribute utility theory applied to generic attributes of the node's context, e.g., the residual battery level, the rate of change of connectivity, the probability of being within reach of the destination, and the degree of mobility. When the best carrier receives a message for forwarding, it stores it in a local buffer and eventually forward it to the destination node when met or, alternatively, to another node with a higher delivery probability. Burgess et al. [12] proposed a protocol called *MaxProp* to schedule transmission of messages to its peers and determine which messages should be deleted when buffer space is almost full. Messages are scheduled based on the path likelihood to peers, which in turn is computed using historical data. In addition, several complementary mechanisms, e.g., acknowledgments, head-starts for new packets, and lists of previous intermediaries are used.

2. *Location-based forwarding*. In the location-based forwarding approaches, nodes will choose the neighbor which moves the data message closest to the destination(s). LeBrun et al. [13] proposed a method using the *motion vector* (MoVe) of mobile nodes to predict their future location. The MoVe scheme uses the knowledge of relative velocities of a node and its neighboring nodes to predict the closest distance between two nodes in future. After the future locations are calculated, messages are passed to nodes that are moving closer to the destination(s). Shen et al. [14] proposed a relay-based routing scheme, called *interrogation-based relay routing* (*IBRR*), for ad hoc satellite networks where nodes are required to buffer data for a certain period of time until the node gets an opportunity to forward it. In IBPR, the nodes interrogate each other to learn more about network topology and node capacity to make intelligent routing decisions. Given the dynamic topology and heterogeneity of an ad hoc satellite constellation, it is very difficult to decide whether or not to forward the data to a given node. To make good routing decisions, satellites use a one-hop look-ahead strategy in which the locations of neighboring nodes and their neighboring nodes are tracked. The parameters being used to make forwarding decisions are spatial and orbital locations of the candidates, bandwidth of the inter-satellite links, relative velocity/mobility between two nodes, vicinity to other satellites and ground stations, capability of the candidates satellites, and data transmission time. Lin et al. [15] propose *indoor geographic routing protocol* (INGEO) for mobile ad hoc networks, in which the velocity displacement vectors are used for relocating the mobile nodes. Basagni et al. [16]

handle client mobility by forwarding messages to all one-hop neighbors located inside a Forwarding Zone (FZ) in the destination's direction. To that end, they proposed *distance routing effect algorithm for mobility* (DREAM), which uses a probabilistic mathematical model to define the FZ, based on the locations of the source and destination(s) and nodes in one-hop neighborhood. The FZ covers a circular region capturing all possible locations of the roaming subscriber since the last location update. Location-based forwarding approaches incur additional overhead while obtaining location updates of their own and neighboring nodes.

3. *Mobility-based forwarding*. The mobile devices in OPMANETs move while following certain known patterns such as walking along a street or driving down the highway. Such regular motion patterns help the intermediate nodes to make accurate estimation of which nodes move toward the destination with higher probability. Leguay et al. [17] proposed a strategy that uses a high dimensional Euclidean space, called *mobility pattern spaces* (*MobySpace*) where each axis represents a possible contact between two nodes, and the distance along an axis measures the probability of that contact to occur. Two nodes that have similar sets of contacts, and that experience those contacts with similar frequencies, are close in the *MobySpace*. The best forwarding node for a message is the node that is as close as possible to the destination node in this space. In this virtual contact space, the knowledge of all the axes of the space also requires the knowledge of all the nodes that are circulating in the space. This full knowledge, however, might not be required for successful routing. Becker et al. [18] proposed *model-based routing* (MBR) that uses world models of the mobile nodes for a better estimation of the locations of the relaying nodes and the receiver, without flooding the network. World models contain location data (e.g., road maps or building charts) and user profiles indicating the motion pattern of users; these location and user profiles are used by MBR to choose a relay that moves toward the target with higher probability. Chen et al. [19] proposed a mobility-based forwarding approach for nodes moving along on a highway. This approach takes advantages of predictable node movement that creates connection opportunities to relay messages in a store-and-forward fashion in an intermittently connected vehicular ad hoc network with frequent network partitions. Messages are propagated greedily each time by forwarding it to the neighbor closest to the destination. Two kinds of transmission schemes are used, i.e., pessimistic forwarding and optimistic forwarding, which are distinguished by how long the messages are allowed to stay buffered at intermediate nodes. To that end, a pessimistic forwarding scheme drops a message when no suitable next-hop node is found. On the other hand, in optimistic forwarding, messages without next hops may remain on intermediate nodes for some time, hoping that physical movement of nodes eventually creates a forwarding opportunity. Jetcheva et al. [20] proposed a realistic approach in which traces from a real-life scenario are used for modeling of user mobility. This approach uses a fleet of city buses as mobile nodes to obtain mobility trace data which is then used to simulate potential latency and routing characteristics, assuming various radio coverage models. Su et al. [21] study the usage of real user mobility and opportunistic pair-wise contacts to form OPMANETs. Their study revealed

that nodes exhibit signs of regularity and affinity of contact. Furthermore, in many cases, success of message delivery from any source to destination is not evenly distributed among the intermediary nodes—this observation can potentially be used by the source nodes for making better routing decisions.

4. *Link asymmetry-based forwarding*. Link asymmetry is a common phenomenon in OPMANETs due to varying transmission ranges, node mobility, heterogeneous radio technologies, and irregularities in radio ranges and path and packet loss patterns. Link asymmetry-based forwarding approaches in OPMANETs exploit asymmetric communication opportunities for shorter delay; however, the cost of discovering asymmetric links in the network is often non-negligible. Sang et al. [22] proposed a novel neighbor discovery technique based on a new one-way link metric in order to identify high reliability forward asymmetric links. They presented a local procedure for their estimation. Duros et al. [23] proposed a modification to the well-known Internal Gateway Protocol OSPF to handle asymmetric satellite links by sharing neighbor tables in Hello packets. Wang et al. [24] proposed a routing protocol for power constrained networks with asymmetric links, where they discover an asymmetric neighbor with the help of a mutual third-party proxy set. Mitra et al. [25] proposed a protocol, called *asymmetric geographic forwarding* (*A-GF*), which discovers and exploits asymmetric links in a network, i.e., instead of eliminating these links, A-GF uses them for routing to reduce the routing hop count (and latency) and to increase the routing reliability when there are no symmetric paths available. A-GF modifies the "Hello message" approach of GF to discover asymmetric links, proactively monitors the changes in conditions of discovered links, and ranks neighbors based on perceived link stability. During routing, a node considers both this ranking and the progress a neighbor can make towards the sink location in terms of geographical distance (thereby considering both reliability and performance). Figure 6.5 shows the asymmetric link discovery and notification procedure in A-GF. In Fig. 6.5a, when node B gets a periodic Hello packet from node A, B checks if it is recorded in the neighbor list in the received packet. If B is listed in A's neighbor list, then both A and B can hear each other and the link A↔B is symmetric. If B is not listed in A's neighbor list, B knows that the link A→B is asymmetric. In A-GF, node A and B are defined as the *up-link node* and the *down-link* node. In general, an asymmetric link is usable in the direction from the up-link node to the down-link node. However, only B is aware of the asymmetric link A→B, but A would not be able to exploit it. Sending a direct (single hop) report from B to A is not possible. Therefore, A-GF routes a small control message, called *asymmetric notification* (*AN*), toward the location of the up-link node A, along a multi-hop *tunneling path* to notify node A of the existence of the outgoing asymmetric link.

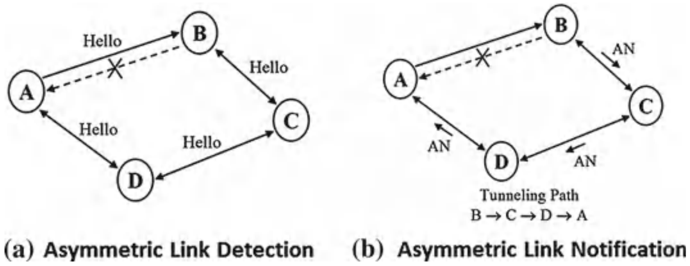


Fig. 6.5 Asymmetric link discovery and notification in A-GF [25]

6.3.2 *Flooding-Based Approaches*

In OPMANETs, applications often require to distribute data among a group of nodes. When the group size is greater than one, the delivery semantics is usually multicast in which the data is intended to be delivered to all members in the same multicast group. Multicasting has often been realized by flooding in wireless networks. The heuristic used by flooding-based algorithms is that data should be diffused all over the network because no knowledge about an appropriate next-hop node is available. Flooding-based techniques work well in highly mobile networks with many communication opportunities between nodes. In addition, such routing approaches result in shorter message delivery delay in highly mobile networks, only at the expense of high resource consumption. To that end, flooding-based approaches suffer from high contention and may potentially lead to network congestion due to large number of data transmissions associated with flooding. Consequently, flooding-based approaches employ a variety of policies to limit flooding, e.g., imposing a maximum number of hops to each data message, or by limiting the total number of message copies present in the network at the same time.

6.3.2.1 Epidemic Routing Approaches

Epidemic routing algorithms constitute a key class of algorithms in OPMANETs that belong to the category of flooding-based approaches. As suggested by the term "epidemic", the routing approaches in this category diffuse data in the network in a fashion similar to diseases or viruses (i.e., by means of pair-wise contacts between nodes). A node is susceptible to infection when it has not yet received the data message. A susceptible node becomes infected when it comes into contact with an infected node and receives the message from it. After catching the infection, the infected node stores the message locally and it is healed when (if at all) it delivers the message to the destination node(s). A healed node also becomes immune to the same disease and does not provide relaying of the same message. Vahdat et al. [26] proposed the first routing algorithm in this category; they showed that the epidemic algorithm ensures that a sufficient number of random exchanges of data in the network guarantee all nodes will eventually receive all messages. Due to their inherent flooding nature, epidemic routing algorithms result in high energy and bandwidth consumption. Vahdat et al proposed to set a bound on the flooding by assigning a hop count limit to each message when it is generated. When the hop count limit is one, the message can only be sent directly to the destination(s). Spyropoulos et al. [6] proposed a technique called *spray-and-wait* to control the amount of flooding. During the spray phase, L copies of the data message are initially spread over the network by the source node or other nodes to L distinct relays. If the destination was not found during the spray phase, each node holding a copy of the message will perform direct transmission during the wait phase. Binary spray-and-wait is another variant of the basic spray-and-wait protocol and produces better performance. In this approach, the source node sends half of the copies of the message to the new relay node, and keeps the other half. The source node and relay nodes repeat this procedure until there is only one copy left, at which point it switches to direct transmission. Su et al. [21] presented an experimental study to test the feasibility of using user mobility and consequent opportunistic pairwise contacts to form an OPMANET inside a college campus. Using commodity mobile devices, they instrumented two user studies for experimentally collecting trace data of user contacts. The approach is unique in that they did not have a predetermined model of user mobility. The results of the experiment are promising, showing that user mobility can potentially be used to form a network. Using this trace data, they simulated an idealized network using epidemic propagation, and observed that nodes exhibited signs of regularity and affinity of contact. The authors further observed that the success of message delivery from any source to destination(s) was not evenly distributed among the intermediate nodes, and they suggested that this fact should be used by source nodes for making better routing decisions.

6.3.2.2 Location-Based Approaches

Geocasting protocols [27–31] are a class of flooding-based protocols that send messages from a source to all hosts located in a specific geographical region. When a node inside the geocasting region receives the messages, it shares the messages with other group members inside the geocasting region by flooding the messages. Group membership changes whenever a mobile node moves in or out of the geocasting region. Leontiadis et al. [32] proposed a VANET-based routing protocol, which allows the publishers to produce events (traffic alerts/notifications) for a particular subscription area/point of interest described by a geographic grid. Multicasting is realized by restricted flooding inside the subscription area. To minimize the number of broadcasts in the subscription area, a small number of event/notification replicas are created, and vehicles carrying those replicas serve as mobile info stations. However, these protocols only consider a special case of multicast where all the subscribers are located inside a specific geographic region. Therefore, they are not suitable for more general cases where the subscribers are arbitrarily located anywhere in the network, as multicasting to the subscribers requires flooding the entire network. Various cluster-based protocols [33–35] have been proposed to alleviate flooding, where the network is partitioned into several disjoint and equally sized cell regions. A cluster manager handles all communications for its cluster, and is responsible for communications with managers of neighboring clusters. Cluster-based protocols thus alleviate flooding, but increase management overheads.

6.3.2.3 History/Estimation-Based Approaches

Burns et al. [36] proposed the *meetings and visits* (MV) routing protocol, which is a further step beyond epidemic routing. Messages are exchanged during pair-wise contacts as in epidemic routing. However, the MV protocol introduces a further complex method to select the messages to forward to a new contact. The choice depends on the probability of previously encountered nodes to successfully deliver messages to their eventual destinations. The delivery probability relies on recent-past observations of both the meetings between nodes and the visits of nodes to geographical locations. Lindgren et al. [37] proposed a similar approach, called *probabilistic routing protocol using history of encounters and transitivity* (PROPHET) that calculates a probabilistic metric for message forwarding. This metric is called delivery predictability and it indicates the probability of successfully delivering a message to the destinations from the local node. PROPHET operates in a similar way as the epidemic routing. When two nodes meet, they exchange summary vectors containing the delivery predictability vector. If two nodes meet very often, they have high delivery predictability to each other. On the other hand, if a pair of nodes does not meet each other in a while, they are intuitively not good forwarders of messages to each other. Hence, the delivery predictability values must age (i.e., be reduced) as time passes. The authors showed in their simulation results that the overhead of PROPHET is lower than that of epidemic routing. Balasubramanian et al. [38] proposed *resource allocation protocol for intentional DTN* (RAPID), in which they presented opportunistic routing as a resource allocation problem. The goal of RAPID is to minimize one of three routing metrics, e.g., average delay, missed deadlines, and maximum delay, using a utility function. This function assigns a utility value, U_i to every packet i, based on the metric being optimized. To that end, U_i is defined as the expected contribution of packet i to this metric. RAPID replicates those packets first which result in the highest increase in utility. For example, if the metric to optimize is average delay, the utility function is defined as $U_i = -D(i)$, i.e., the negative of the average delay. Hence, the protocol replicates the packet that results in the greatest decrease in delay. Due to the inherent flooding nature, RAPID attempts to replicate all packets if network resources are available.

6.3.2.4 Network Coding-Based Approaches

Flooding-based algorithms also include network coding-based approaches [39] that take an original approach to limit message flooding. Just to give a classical example, assume A, B, and C are the only three nodes of a small network. Let us assume that

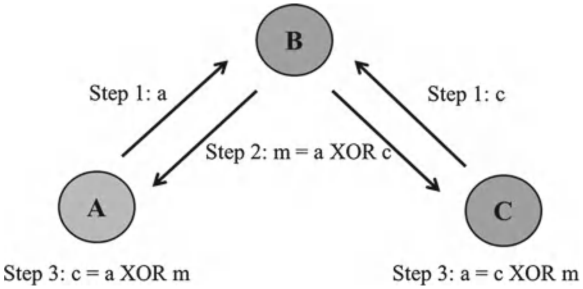


Fig. 6.6 Example of network coding-based forwarding

node A generates the data message "*a*" and node C generates the data message "*c*." In order to distribute their data to all other nodes, nodes A and C send their messages to node B. Instead of sending two separate data messages for "*a*" and "*c*," respectively, node B broadcasts a single message containing "*a*" XOR "*c*." Once "*a*" XOR "*c*" is received, both nodes A and C can finally infer the data messages sent by the other source (i.e., node A can infer "*c*" and node C can infer "*a*"). Network coding-based routing approaches outperform basic flooding, as they are able to make data delivery with a fewer number of messages injected into the network.

6.3.3 Hybrid Approaches

As stated above, flooding-based approaches work well in highly mobile networks with many communication opportunities between nodes arising due to node mobility. However, these approaches generate high overhead, network congestion and may suffer from high contention. Several hybrid approaches have been proposed in recent literature, which combine features from forwarding-based approaches with flooding-based approaches to achieve low data dissemination latency while limiting the network overheads. To that end, a forwarding-based approach is used to get the data message as close to the destinations as possible. Once the data message gets close to the destinations, flooding-based approaches are used to disseminate the data in the surrounding. Tchakountio et al. [40] proposed the *Spray* routing protocol, which is an integrated location tracking and forwarding scheme. The idea behind spray routing is that even though a highly mobile node may not be in the location last reported by the location tracking mechanism, it is likely to be in one of the surrounding locations. Consequently, by "spraying" the data message to the vicinity of the destination's last-known location, the algorithm attempts to deliver packets to the destination even if it moves to a nearby location during the location tracking convergence time. The data packet is first unicast along a ray to a node in the vicinity of the destination's last known location before being multicast in that vicinity area. The magnitude of the spraying depends on the mobility; the higher the mobility, the larger the vicinity. Nain et al. [41] proposed a relay-based approach, called *mobile* *relay protocol* (MRP) that is used in conjunction with epidemic style broadcasting. To that end, MRP integrates message routing and storage in the network. The basic idea is that if a route to a destination is not available, a node performs a controlled local broadcast to its one-hop neighbors. All neighbors receiving this broadcast store the data message and enter the relaying mode. In the relaying mode, the MRP protocol first checks its routing table to see if an end-to-end route of less than d hops exists. If so, the data message is forwarded to the destination(s). If no valid route is found, the MRP protocol enters the storage mode, which involves the following steps: if the message is already stored in the current node's buffer, then the older version of the message is discarded and the new version is stored given the buffer is not full. In case the buffer is full, the MRP protocol removes the least recent message from the buffer and relays to a single *random* neighbor. In a network with sufficient mobility, it is quite likely that one of the relay nodes to which the packet has been relayed will encounter a node that has a valid route to the destination, thereby increasing the likelihood that the message will be successfully delivered. Mitra et al. [42] proposed a hybrid overlay multicast protocol for OPMANETs, called *Courier* that creates potential overlay multicast trees using the location and velocity of the source node and destinations. The construction of the overlay multicast tree is guided by a mobility prediction model. Each node is represented by a circular region called *node forwarding zone* (NFZ); the center of the NFZ is the last-known location of the node and its radius is equal to the last-known velocity of the node times the time duration since last location update. The NFZ, by definition, captures all possible locations in all directions to which the mobile destination could have moved to at its last-known velocity. During multicast, once the data message enters the NFZ of a destination, it is broadcasted by all nodes inside that circular region. Thus, Courier ensures that the data message is delivered to the mobile destination even if it is no longer at its last-known location. Courier reduces the overlay multicast group management cost by using a bandwidth efficient location update scheme. Between two overlay nodes in the multicast tree, the data message is forwarded using a location-based forwarding protocol that exploits transient communication opportunities between mobile nodes.

6.3.4 Social Behavior-Based Approaches

Exploiting social behavior for increasing routing efficiency in OPMANETs has recently received considerable attention in research community. To that end, social behavior-based approaches use social-based metrics derived from contacts between devices to make opportunistic forwarding decisions with low overhead. Many approaches implicitly assess the strength of "social" ties between nodes, using metrics such as time of last encounters between nodes [43] or frequency of contacts as a hint of similarity between mobility patterns [44]. However, these simple metrics may only capture one facet of the underlying mobility patterns, which can in turn hinder good contact predictions. Complex network analysis is a further generic and powerful tool to formulate and solve the problem of future contact prediction. Past

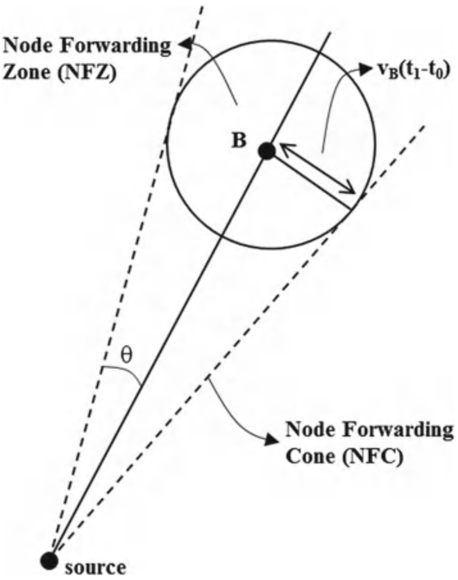


Fig. 6.7 Example of mobility prediction-based hybrid multicast in Courier [42]

observed contacts between nodes are aggregated into a social graph, with graph edges representing past meetings between the vertices. Two nodes who often meet either have a strong social tie (friends), or they are frequently co-located without actually knowing each other (familiar strangers); thus, existence of an edge intends to have predictive capacity for future contacts. Protocols using such explicit knowledge of social relationships have been proposed to improve efficiency over socially agnostic protocols in OPMANETs. Hui et al. [45] proposed a routing protocol, called *Bubble Rap* that uses the patterns of contacts between users/nodes to classify the users into various distinct social communities, which in turn are used as context for routing. Bubble Rap gives preference to nodes belonging to the same community of the destinations as good forwarding candidates. If such nodes are not found, Bubble Rap forward the message to the nodes that have more chances of contact with the community of the destinations. Daly et al. [46] proposed *SimBet* that assesses "similarity" (i.e., the number of neighbors two nodes have in common) to detect nodes that are part of the same community, and "betweenness centrality" (the fraction of shortest paths between each possible pair of nodes going through this node) to identify bridging nodes, that could carry a message from one community to another. The decision to forward a message depends on the similarity and centrality values of the newly encountered node, relative to the current one. If the former node has a higher similarity with the destination, the message is forwarded to it; otherwise, the message stays with the most central node. The goal is to first use increasingly central nodes to carry the message between communities, and then use similarity to "home in" to the destination's community. Mtibaa et al. [47] proposed *PeopleRank* in which nodes are ranked using a tunable weighted social behavior. Similar to the PageRank idea, PeopleRank gives higher weight to nodes if they are socially connected to other high profile nodes of the network. The authors developed centralized and distributed variants for computing the PeopleRank metric and used real mobility traces of nodes and their social interactions for evaluation. PeopleRank delivers messages with near optimal success rate close to epidemic routing, while reducing the number of message retransmissions by 50 % compared to epidemic routing. These approaches identify “strong social ties” among mobile devices, inferred either from contact behavior or declared friendship. Zyba et al. [48] explored the role and potential of non-social, vagabond devices for communication and data dissemination. They used experimental traces to study fundamental properties of human interaction; the traces were broken down in multiple areas and mobile users in each area were classified according to their social behavior. Socials are devices that show up frequently or periodically, while Vagabonds represented the rest of the population. Their study of the traces revealed that in some situations (specifically, beyond a "tipping point"), population size had a greater impact on the effectiveness of data dissemination than the social status of devices. To that end, an analytic model was proposed that formally characterizes the relationship between population size and the social behavior of users, and identifies a simple formula for determining when data dissemination through Vagabonds outperforms dissemination through Socials.

Summary: Section 6.3 classified routing protocols in OPMANETs into four different categories: forwarding-based, flooding-based, hybrid, and social behavior-based. The forwarding-based approaches were further divided in two categories: direct transmission-based and context-based. Table 6.2 summarizes the various categories of routing protocols, using the analytical framework proposed in Sect. 6.2.

6.4 Real-Life Applications of Opportunistic Networks

In OPMANETs a network is typically separated into several network partitions called regions. Traditional applications are not suitable for such environments because they normally assume an end-to-end connection from the source to the destination(s). The applications that are suitable for OPMANETs are typically asynchronous in nature and tolerant of long delay and high error rates. This section discusses in detail of a few applications of OPMANETs and specifically provides insights on the routing approaches used by these applications.

6.4.1 Urban Environments

The Haggle project[2] was a four-year project, started in January 2006 and funded by the European Commission[3] and was targeted at providing solutions for communications in OPNETs. Haggle performed an extensive study of the properties of pocket

---

[2] http://www.haggleproject.org.

[3] http://cordis.europa.eu/ist/fet/comms-sy.htm.

Table 6.2 Summary of the OPMANET routing protocols

Routing approach	Analysis framework	Features
Direct transmission-based approaches	Dissemination	Direct transmission-based approaches minimize the number of data message transmissions by limiting the number of hops traveled by the data message between source and destination(s)
	Resource usage	They minimize transmit power consumption; but they assume infinite storage space which is not realistic for mobile devices in OPMANETs
	Applications	The assumption of complete mixing of node trajectories so that every node can get close to every other node in the network limits the applications enabled by this category of routing protocols. As a result, these approaches have the fewest number of real-life applications
Context-based approaches	Dissemination	These approaches exploit the context (e.g., historical data, location, mobility pattern etc.) in which nodes are operating so as to identify suitable next hops toward the eventual destinations
	Resource usage	They generate more number of copies of a message in the network but achieve shorter. Moreover, these approaches have higher computational and storage overheads because nodes need to maintain states about other nodes in the network
	Applications	They enable a variety of asynchronous, delay-tolerant applications in which some kind of context information about participating nodes are available
Flooding-based approaches	Dissemination	Flooding-based approaches work well in highly mobile networks with many communication opportunities between nodes arising due to node mobility. The most common dissemination method is epidemic routing
	Resource usage	They generate high overhead, network congestion and may suffer from high contention. Approaches usually differ from one another in the methods they employ to limit flooding overheads
	Applications	These approaches are simple to implement and have the highest number of real-life applications

(continued)

Table 6.2 continued

Routing approach	Analysis framework	Features
Hybrid approaches	Dissemination	These approaches combine features from forwarding-based approaches with flooding-based approaches. First, a forwarding-based approach is used to get the data message as close to the destination as possible. Next, the data is disseminated in the surrounding using a flooding-based approach
	Resource usage	Hybrid approaches achieve low data dissemination latency while limiting the overheads
	Applications	These approaches enable asynchronous applications that require multicast/group-based communication semantics
Social behavior-based approaches	Dissemination	These approaches use social-based metrics derived from contacts between devices to make opportunistic forwarding decisions. Such approaches implicitly/explicitly assess the strength of "social" ties between nodes, using metrics such as time of last encounters between nodes, frequency of contacts as a hint of similarity between mobility patterns, or complex network analysis
	Resource usage	They have low communications overhead but incur high computational and storage overheads because they need to maintain states about encountered nodes to assess their social behavior
	Applications	These approaches often use real-life mobility traces to make forwarding decisions and as a consequence, result in very high contact prediction accuracy. Suitable for urban applications with high user mobility and regular mobility patterns

switched networks (PSNs), i.e., OPNETs that can exploit any contact opportunities between devices (e.g., cell phones and PDAs that users carry in their pockets) to forward messages. The project put special emphasis on measuring and modeling pair-wise contacts between devices. To that end, pair-wise contacts between devices were characterized by the means of two parameters: contact durations and inter-contact times. The duration of a contact is the total time that two tagged mobile devices are within reach of each other. On the other hand, the inter-contact time is defined as the time in between two contact opportunities between the same two tagged devices. While the contact duration directly influences the capacity of OPNETs because it limits, the amount of data that can be transferred between nodes, the inter-contact time influences the feasibility of OPNETs and the delay associated with them. To characterize contact durations and inter-contact times occurring in real-world environments, various sets of traces, e.g., logs collected by the APs of university campuses, by devices carried by students and researchers in their university and laboratories,

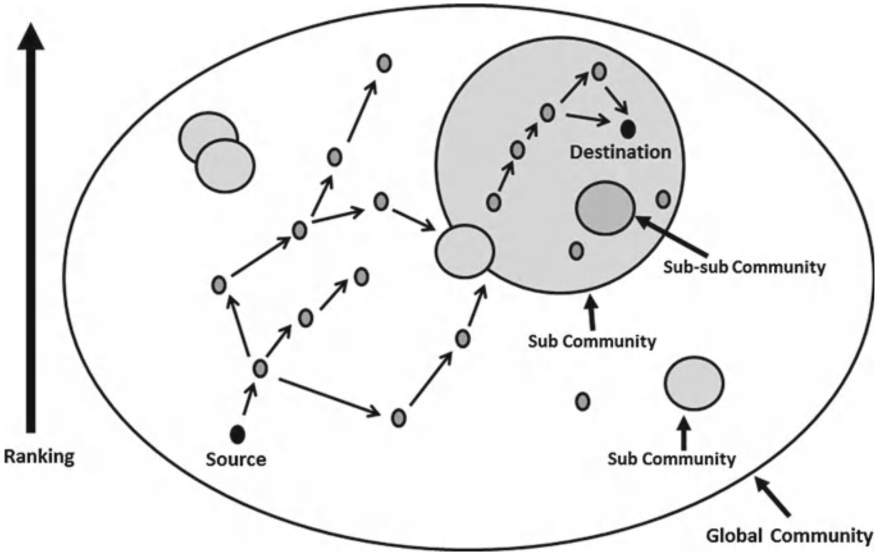


Fig. 6.8 Example of social community-based forwarding in Bubble Rap [45]

and by the participants to some international conferences, were collected and analyzed. The analysis of all the traces had led to the conclusion that both inter-contact times and contact durations are characterized by heavy-tailed distribution functions that approximately follow power laws. The Haggle architecture enables a social and context-aware content-sharing service that exploits a context definition designed for OPNETs. The service uses the users' social behavior patterns to identify content that might be relevant to the communities a user interacts with. More specifically, these interaction patterns allow the system to improve forwarding decisions by probabilistically predicting future user contacts.

Heinemann et al. [1] proposed adPASS, an OPNET framework that disseminates advertisements among interested users in an urban environment, using a word-of-mouth recommendation style approach. For example, assume Alice carries a mobile device with her. A personal profile, stored on her device, stores her interests and knowledge. The device is able to match her profile with other nearby devices by communicating wirelessly and without user interaction. On the other hand, a shop has put a fixed device next to the shop window. This device announces digital advertisements from the shop to passersby. As Alice passes the shop window, her device learns about the special offer for digital cameras. Alice physically carries the advertisement with her and passes it further to other users she encounters. All users interested in the ad (including herself and her colleague Bob) might take the chance and visit the shop in order to buy the advertised product. Data dissemination relies solely on one-hop communications and uses a node's profile to carry out its task. When two nodes detect a match in their node profiles, they exchange data. The physical movement of nodes is utilized to distribute the data. adPASS also makes use of Information Sprinklers for implementing time-shifted information pass, i.e., users who are at the same place but at a different time are able to share information with one another. As an example, consider a user Alice who goes to a local coffee bar at 10 every morning. User Bob visits the same place each afternoon. Alice and Bob will never meet and thus come into communications range while visiting that coffee bar. In this situation, an Information Sprinkler installed in the bar can collect all information of all users visiting the bar. This allows Alice to leave her information at the Sprinkler in the morning and Bob to learn about this information from the Sprinkler in the afternoon.

6.4.2 Monitoring

Wildlife monitoring is another possible domain of applications of OPMANETs. It focuses on tracking wild species to understand their behavior, interaction, and mutual influence on one another, and as well as their reaction to the ecosystem changes caused by human activities. Researchers use OPMANETs as a reliable, cost-effective, and non-intrusive means to monitor large population of animals roaming in vast regions. Systems for wildlife monitoring generally include special tags with sensing capacity to be carried by the animals under study, and one or more base stations to collect the data from the tags and send them to the processing center. Base stations can be fixed or mobile; however, in both cases data collection from all the deployed tags is quite challenging. As a consequence, routing protocols exploit pair-wise contacts between the animals to let them exchange the data already collected. Eventually, each animal carries the data collected on its own and the data collected by other animals it has encountered. ZebraNet [49] is an interdisciplinary ongoing project at Princeton University and its deployment scenario is the vast savanna area of central Kenya. The animals to be tracked are zebras wearing special collars. The base station consists of a mobile vehicle for the researchers, which periodically moves around in the savanna and collects data from the zebras encountered. Two alternative protocols have been considered for data collection in ZebraNet. The first one is simple flooding-based; each collar sends all its data to each encountered neighbor until the data eventually reaches the base station. The second one, named history-based protocol, proposes that each node selects only one of its neighbors as relay for its data. The selected node is the one with the highest probability to eventually encounter the base station. Each node is thus assigned a hierarchy level (initially zero) that increases each time it encounters the base station, and conversely decreases after not having seen the base station for a certain amount of time. When sending data to a relay node, the neighbor to be selected is the one with the highest hierarchy level. Simulation results show that both forwarding protocols outperform the direct protocol, in which each collar has to directly communicate with the base station to upload data. Moreover, the history-based protocol outperforms flooding in terms of bandwidth and energy consumption. Small et al. [50] proposed the *shared wireless infostation model* (SWIM) that monitors whales using special tags. Data is replicated and diffused at each pair-wise contact between whales (similar to the flooding protocol of ZebraNet) and finally arrives to special SWIM stations that can be fixed (on buoys) or mobile (on seabirds).

Hence, both whale-to-whale and whale-to-base station communications are allowed. From the SWIM stations, data messages are eventually forwarded onshore for final processing and utilization.

Pompili et al. [51] proposed a geographic routing solution for *delay-insensitive underwater sensor network applications*. Such networks consist of sensors and vehicles deployed to perform collaborative monitoring tasks over a given underwater region for a variety of applications, e.g., oceanographic data collection, pollution monitoring, offshore exploration, disaster prevention, assisted navigation, tactical surveillance, and mine reconnaissance. The proposed solution aims to efficiently exploit the underwater acoustic channel and minimize the energy consumption. The protocol increases the efficiency of the channel by transmitting a train of short packets back-to-back and limiting the packet error rate by keeping the transmitted packets short. Each node is allowed to select its best next-hop node, the transmission power, and the forward error correction code rate for each packet. Furthermore, it tries to exploit those links that guarantee a low packet error rate, in order to maximize the probability that the packet is correctly decoded at the receiver. For these reasons, the energy efficiency of the links is weighted with the number of retransmissions required to achieve link reliability, with the objective of saving energy.

6.4.3 Connecting Developing Areas

OPMANETs can be used to provide Internet connectivity to rural and developing areas where they are typically the only affordable way to help bridge the digital divide. One such example is the *DakNet Project* [52] which is aimed at realizing a very low-cost asynchronous ICT infrastructure so as to provide connectivity to rural villages in India. To that end, kiosks are built up in villages and equipped with digital storage and short-range wireless communications. Periodically, mobile access points (MAPs) mounted on buses, motorcycles, or even bicycles pass by the village kiosks and exchange data with them wirelessly. MAPs can download data stored at the kiosks, and upload them to the Internet when passing by an access point (AP) in a nearby town. Similarly, MAPs may download, from the Internet, the requested data and bring it to villages. DakNet has the potential to support Internet/Intranet messaging (e.g., email, audio/video messaging, and mobile e-commerce), data distribution (e.g., public health announcements, community bulletin boards, news, and music), and collection (e.g., environmental sensing, voting, health records, and census). A similar model is used to implement a hybrid real-time and store-and-forward Wi-Fi mesh network in Kigali, Rwanda.[4] As part of a hybrid *village area network* (*VAN*) implemented to provide real-time Internet access at various sites throughout the capital of Kigali, a unique store-and-forward connectivity is demonstrated in the countryside. Throughout all the implementation, major emphasis is put in the formation and training of a local technical team in order to ensure the proper transfer

---

[4] http://www.firstmilesolutions.com/projects.php?p=rw-dk.

of technology and its associated knowledge. Eventually, the local team is capable of operating, troubleshooting, and expanding the network at will. The *Saami network connectivity* (*SNC*) project [53] aims to provide network connectivity to the nomadic Saami population of the reindeer herders. Saami herders live in the northeast part of Sweden, Norway, and Finland and move from their villages through the year following the migration of reindeers. Providing network connectivity to the Saami population is a means to protect and defend their habitats, culture, and traditions while also supporting their integration into the modern society of their countries. The network connectivity enables the Saami to achieve economic growth through distance work and Internet-based business. Network-based services can also allow Saami children to receive their education without the need to leave their parents to attend boarding schools. Network connectivity can also give Saami more visibility, and let them have more influence in the political and economic affairs of their country. In its initial stage, the SNC project has only focused on providing email, file transfer, and cached web services to the Saami people. Reindeer herd telemetry is also going to be provided to support the herding activity itself.

6.5 Discussion

Based on the existing researches and the unique characteristics of OPMANETs, there are still a few research issues that are far from being adequately addressed. To that end, this chapter found a few gaps in many of the current OPMANET routing approaches that will pose significant challenges in future as more applications targeted at OPMANETs come into existence. The key observations made in this chapter are listed below.

6.5.1 Advanced Personal Mobile Devices Should be Used for Increasing Network Capacity

The rapid proliferation of sophisticated mobile devices (e.g., smartphones and tablets) equipped with powerful processors, abundant memory, and an ever-increasing number of sensors (e.g., to capture user location, audio/video, ambient light, etc.) presents the potential of building large-scale dynamic OPMANET applications. These devices are suited for large-scale OPMANET applications because they are always mobile, which makes it very likely that there will be devices at the right place at the right time to successfully accomplish the required forwarding task, thereby increasing the network capacity.

6.5.2 Use of Heterogeneous Devices and Radio Technologies Should be Encouraged for Increasing the Number of Communication Opportunities in the Network

One major limitation in all approaches discussed in this chapter is that developers are currently tied to the specifics of one particular wireless communications technology. However, the number of devices with diverse radio communications technologies is increasing (e.g., smartphones and tablets are equipped with multiple radio technologies such as Wi-Fi, Bluetooth, and RFID) and one should exploit such heterogeneous communication opportunities for increased data delivery success in OPMANETs. A solution targeted at providing this type of interoperability should focus on developing a technology-independent, high-level software application programming interface (API) for OPMANET applications. Given that many of these heterogeneous devices use heterogeneous operating systems with some of them being proprietary, the obvious level for implementing such API is at the middleware level. Such an API will not only increase the usage lifetime of the applications but also encourage the creation of new applications by helping to amortize development efforts.

6.5.3 Privacy and Security

OPMANET applications communicate with their surroundings without active user interaction, and this raises many privacy issues with increasing use of personal mobile devices in urban environments. These devices store personal data and interests and thus, while participating in data forwarding, they become vulnerable to the danger of tracking and monitoring user behavior for constructing user profiles and compromising user anonymity in the network. Current applications that address the issue of protecting user privacy in OPMANETs either focus on protecting location privacy [54, 55] or are heavily tailored to the respective applications [56, 57]. Future research efforts should focus on developing generic privacy and user identity protection techniques targeted at OPMANET applications. In addition, new security and data encryption mechanisms should be developed for devices with limited resources working in a dynamic and distributed environment, as techniques that rely on access to a centralized service cannot be used, or the assumption that all the intermediate nodes are trustworthy is no longer valid.

6.5.4 Applications Should be able to Specify Their Quality of Service Requirements

In OPMANETs, both source and destinations frequently switch states between connection and disconnection to and from the network; as a result, it is very challenging to ensure reliable data delivery in such environments. To that end, the loosely coupled and asynchronous nature of OPMANETs introduces a non-deterministic behavior in the system, thereby making it very hard to provide QoS guarantees. Future research approaches should aim at enabling the applications to specify their own QoS requirements to the routing protocol, which in turn should include approaches for meeting those requirements. For example, applications can specify that they require their multicast data to be persistent, thus ensuring that the messages will not be lost in the event of a source node failure. Persistent data delivery could be implemented using policies such as durability of subscription and multicast data storage and retransmission. To that end, routing protocols could require the source node to log messages in a stable storage. On the other hand, durability of subscription will allow destinations to receive the messages sent while they were disconnected from the network. This will require implementing reliable queues on part of the source nodes—when a destination that was disconnected rejoins the network and the durability of its subscription is still valid, the source node resends the stored messages to the client. In order to minimize duplicated messages and reduce the associated communications overhead, routing protocols could attach an expiration time to each multicast data message.

6.5.5 Node Location and Velocity Should be Used by Routing Approaches

Although many of the current OPMANET routing approaches already use node location, only a handful of them use both location and velocity for making mobility prediction. To that end, the forwarding approaches should leverage node location and velocity with simple and accurate link availability estimation algorithms for making intelligent forwarding decisions. For example, if a contact is not going to be available until the data message transfer is finished, the routing protocol should store the message in buffer and wait for future communication opportunities, thereby saving on bandwidth and energy resources.

6.5.6 Social Behavior-Driven Traces Should be Used for Deriving Realistic Mobility Models

Since OPMANETs exploit users' mobility to bridge disconnections and partitions in the network, it is of high importance to identify realistic mobility models, both to drive the protocols' design, and to provide sensible performance results. Mobility modeling for OPMANETs is emerging as a hot topic since last few years. To that end, popular models used in MANET research (e.g., the random waypoint model) generate unrealistic user behavior. To address this problem, mobility models are being reconsidered or redesigned based on real users' mobility traces. A few research projects have found that mobility traces can be strongly associated with sociological-inspired concepts, e.g., periodic association patterns follow sociological orbits defined by the social behavior and relationship of users. Such social-aware mobility models have been shown to closely reproduce statistical features of real movement traces, and are thus very good candidates for designing and evaluating OPMANET routing approaches. Future research efforts should focus on analyzing mobility traces from a variety of diverse real-life scenarios for developing realistic social-aware mobility models.

6.5.7 Hybrid Routing Protocols Should be Further Investigated for Increasing Data Dissemination Efficiency in OPMANETs

Hybrid routing protocols combine features from forwarding-based approaches with flooding-based approaches to achieve low data dissemination latency while limiting the network overheads. Although Sect. 6.3.3 discusses a few such protocols, we feel that this class of routing approaches has not been adequately investigated. Future research efforts should focus on developing hybrid approaches that can dynamically mix-and-match approaches from the set of forwarding-based approaches with the ones from the set of flooding-based approaches, as the multicast group size and resources capabilities of the nodes vary in the network. For example, hybrid approaches typically flood the data message once it gets closer to the destinations. The size of the flooding region varies depending on a number of factors (e.g., node mobility, the time duration since last location update from destinations, etc.); if the flooding region becomes too big, then the hybrid routing protocol should dynamically switch from the basic flooding scheme to a history-based flooding scheme that will predict the region where the mobile destination is likely to be found, thereby minimizing the flooding overhead.

6.5.8 Communication Opportunities can be Artificially Created

This chapter focused on the routing protocols that exploit only those communication opportunities that arise spontaneously when nodes come into contact due to node mobility, changes in transmission power, etc. On the other hand, there is another class of routing protocols targeted at sensor networks (which are beyond the scope of this chapter); this class of routing approaches opportunistically employs additional mobile nodes to offer a message relaying service as message ferries [58, 59]. These ferry nodes move around in the network following a predefined trajectory and collect messages from source nodes. To that end, a source node wishing to deliver data messages sends a request message to the ferry with its current position and then the ferry changes its trajectory to meet up with source node. These ferry-based routing protocols artificially create communication opportunities in a sensor network, and they can be further investigated to draw insights to be used in other application scenarios in OPMANETs, e.g., in an urban scenario, buses that follow a predefined trajectory can be used as message ferries.

6.5.9 Multi-Tier Routing Protocols can Increase Network Connectivity at Scale

This chapter discussed routing approaches that focus on a single-tier OPMANET. However, as suggested by Pelusi et al. [3], future routing approaches should investigate the feasibility of building multi-tier OPNETs, in which each tier of the network is an OPNET in itself with its own routing protocol that provides communications among nodes in that tier. Nodes would rely on the upper tiers to reach nodes that are too far away, thus enabling connection among various groups of disconnected lower tier nodes in a sparse/partitioned network. The data mule-based routing approach [60], originally targeted at building multi-tier sensor networks, can provide meaningful insights about building multi-tier OPMANETs for increased connectivity in large-scale systems.

6.5.10 User Participation Should be Rewarded to Increase Message Forwarding

Most of the current OPMANET applications typically share a common goal they want to accomplish (e.g., wildlife/environmental monitoring, providing Internet connectivity to developing areas, etc.). With the increasing use of personal mobile devices (e.g., smartphones, tablets) for implementing OPMANET applications, the assumption of an altruistic resource sharing environment is no longer feasible. These personal user devices serve a variety of strictly personal purposes, e.g., surfing the web, playing games, running other applications, storing calendar, schedules, and contact lists, etc. Since battery, storage, and processing power are limited and precious resources on such personal user devices, OPMANET applications should provide means to reward participating users that help in forwarding/disseminating the data messages. A few recent research efforts [1, 61, 62] have proposed incentive-based solutions for increasing user participations in OPMANETs; these solutions are similar to the incentive-based schemes used by peer-to-peer file sharing systems. Future research efforts should investigate other approaches for rewarding user participations in OPMANETs.

6.6 Conclusion

This chapter provides an overview of the emerging research area of routing in OPMANETs. While this chapter has not covered all aspects of research or all projects in this area, it attempts to provide insights in this evolving area by characterizing a few key research projects using a simple analytical framework—this framework characterizes the routing approaches in terms of three aspects, i.e., data dissemination, resource management, and applications. The framework reveals that the flooding-based approaches work best in highly mobile networks in which many communication opportunities between nodes arise due to node mobility. These approaches are simple to implement and have the highest number of real-life applications; however, they generate high networking overhead, network congestion and contention, etc. On the other hand, hybrid approaches achieve low data dissemination latency while limiting the networking overheads but they are only suitable for applications that require multicast/group-based communication semantics. With the recent growth in use of smartphones and tablets, the social behavior-based approaches are getting a lot of attention. These approaches use real-life mobility traces to make forwarding decisions. As a consequence, they result in very high contact prediction accuracy and low communications overhead. However, they incur high computational and storage overheads because they need to maintain state information of the encountered nodes in order to assess their social behavior.

This chapter also outlines a few gaps in the recent research efforts and provides insights on how to address those gaps in future work. Lack of interoperability is one of the main shortfalls in the existing OPMANET routing approaches. The accelerating global popularity of advanced mobile devices boasting powerful processors, abundant memory, and complex sensor capabilities has opened the gateway to building and deploying large-scale OPMANET applications anytime, anywhere. In order to be able to completely exploit the benefits offered by these widely available, sophisticated mobile devices which often come with heterogeneous platforms and radio communications technologies, we need to provide interoperability between the routing approaches designed for specific platforms/wireless technologies. Future research should investigate the feasibility of developing high-level software API for providing interoperability to OPMANET applications—such an API has the capability of not only increasing the usage lifetime of the applications but also encourage the creation of new applications by helping to amortize development efforts. Another important aspect of deploying collaborative OPMANET applications on personally owned mobile devices is that we need to provide means to reward the participating users—these users share the resources of their personal devices for a common goal that the OPMANET applications want to accomplish, e.g., wildlife/environmental monitoring, providing Internet connectivity to developing areas, etc. As a final point, Lilien et al. [63] argues that OPMANETs are one possible approaches of ultimately moving toward the goal of pervasive computing, and hence it will be increasingly important for OPMANET routing solutions to ensure privacy protection on the consumer mobile devices. Future solutions should investigate different approaches for privacy protection, e.g., encryption of private data and resources, use of adaptive sharing policies, etc.

The future research efforts in routing in OPMANETs will require drawing on expertise from a number of disciplines including (but not limited to) wireless sensor networks, social sciences, network security, algorithm design and evaluation, distributed file systems, e-commerce, etc. Our hope is that this book chapter will stimulate discussion and aid research in this area.

References

1. Heinemann A (2007) Collaboration in Opportunistic Networks. Ph.D. dissertation, Technical University of Darmstadt
2. Zhang Z (2006) Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges. IEEE Commun Surveys Tutorials (1st Quarter) 8(1):24–37
3. Pelusi L, Passarella A, Conti M (2006) Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. IEEE Commun Mag 44:134–141
4. Huang C, Lan K, Tsai C (2008) A survey of opportunistic networks. In: Proceedings of 22nd international conference on advanced information networking and applications. Gino-wan City, Okinawa, Japan, March
5. Phanse K, Nykvist J (2006) Opportunistic wireless access networks. In: Proceedings of 1st international conference on access networks, Greece, Athens, September
6. Spyropoulos T, Psounis K, Raghavendra CS (2004) Single-copy routing in intermittently connected mobile networks. In: Proceedings of 1st IEEE communications society conference on sensor and ad-hoc communications and networks, Santa Clara, CA, USA, October
7. Grossglauser M, Tse DNC (2002) Mobility increases the capacity of ad-hoc wireless networks. IEEE/ACM Trans Networking 10(4):477–486
8. Jain S, Fall K, Patra R (2004) Routing in delay tolerant network. In: Proceedings of the annual conference of the ACM special interest group on data communication (SIGCOMM), Portland, OR, USA, August
9. Jones EPC, Li L, Ward PAS (2005) Practical routing for delay tolerant networks. In: Proceedings of the annual conference of the ACM special interest group on data communication (SIGCOMM), Philadelphia, PA, USA, August
10. Tan K, Zhang Q, Zhu W (2003) Shortest path routing in partially connected ad hoc networks. In: Proceedings of the IEEE global telecommunications conference (GLOBECOM), San Francisco, CA, USA, December
11. Musolesi M, Hailes S, Mascolo C (2005) Adaptive routing for intermittently connected mobile ad hoc networks. In: Proceedings of the 6th IEEE international symposium of world of wireless, mobile and multimedia networks (WoWMoM), Taormina-Giardini Naxos, Sicily, Italy, June
12. Burgess J, Gallagher B, Jensen D, Levine BN (2006) MaxProp: routing for vehicle-based disruption-tolerant networks. In: Proceedings of the annual IEEE conference on computer communications (INFOCOM), Barcelona, Spain, April
13. LeBrun J, Chuah CN, Ghosal D, Zhang M (2005) Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks. In: Proceedings of the IEEE vehicular technology conference (VTC), Montreal, Canada, September
14. Shen C, Borkar G, Rajagopalan S, Jaikaeo C (2002) Interrogation-based relay routing for ad hoc satellite networks. In: Proceedings of the IEEE global telecommunications conference (GLOBECOM), Taipei, Taiwan, November
15. Lin H, Labiod H (2006) INGEO: indoor geographic routing protocol for MANETs. In: Proceedings of the 3rd international conference on mobile computing and ubiquitous networking, San Jose, CA, USA, July

16. Basagni S, Chlamtac I, Syrotiuk VR (1998) A distance routing efficient routing algorithm for mobility (DREAM). In: Proceedings of the 4th ACM/IEEE international conference on mobile computing and networking, Dallas, TX, USA, October
17. Leguay J, Friedman T, Conan V (2006) Evaluating mobility pattern space routing for DTNs. In: Proceedings of the annual IEEE conference on computer communications (INFOCOM), Barcelona, Spain, April
18. Becker C, Schiele G (2001) New mechanisms for routing in ad hoc networks. In: Proceedings of the 4th plenary cabernet workshop, Pisa, Italy, October
19. Chen Z, Kung HT, Vlah D (2001) Ad hoc relay wireless networks over moving vehicles on highways. In: Proceedings of ACM symposium on mobile ad hoc networking and computing (MOBIHOC), Long Beach, CA, USA, October
20. Jetcheva JG, Hu YC, Pal Chaudhuri S, Saha AK, Johnson DB (2003) Design and evaluation of a metropolitan area multitier wireless ad hoc network architecture. In: Proceedings of the 5th IEEE workshop on mobile computing systems and applications, Monterey, CA, USA, October
21. Su J, Chin A, Popivanova A, Goel A, Lara E (2004) User mobility for opportunistic ad-hoc networking. In: Proceedings of the 6th IEEE workshop on mobile computing systems and applications, Lake District National Park, United Kingdom, December
22. Sang L, Arora A, Zhang H (2007) On exploiting asymmetric wireless links via one-way estimation. In: Proceedings of the 8th international symposium on mobile ad hoc networking and computing (MOBIHOC), Montreal, Canada, September
23. Duros E, Dabbous W (1996) Handling of unidirectional links with OSPF. Internet Draft (http://tools.ietf.org/html/draft-ietf-ospf-unidirectional-link-00), March
24. Wang G, Ji Y, Turgut D (2004) A routing protocol for power constrained networks with asymmetric links. In: Proceedings of the 1st international workshop on performance evaluation of wireless ad hoc, sensor, and ubiquitous networks, Venice, Italy, October
25. Mitra P, Poellabauer C (2011) Asymmetric geographic forwarding: exploiting link asymmetry in location aware routing. Int J Embed Real-Time Commun Syst (IJERTCS) 2(4):46–70
26. Vahdat A, Becker D (2000) Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University
27. Boleng J, Camp T, Tolety V (2001) Mesh-based geocast routing protocols in an ad hoc network. In: Proceedings of the international workshop on parallel and distributed computing, San Francisco, CA, USA, April
28. Jiang X, Camp T (2002) A review of geocasting protocols for a mobile ad hoc network. In: Proceedings of the grace hopper celebration of women in computing, Vancouver, British Columbia, Canada, October
29. Ko Y, Vaidya NH (2000) GEOTORA: a protocol for geocasting in mobile ad hoc networks. In: Proceedings of the 8th international conference on network protocols, Osaka, Japan, November
30. Liao WH, Tseng YC, Lo KL, Sheu JP (2000) Geogrid: a geocasting protocol for mobile ad hoc networks based on grid. J Internet Technol 1(2):23–32
31. Stojmenovic I (1999) Voronoi diagram and convex hull based geocasting and routing in wireless networks. Technical Report, University of Ottawa, tr-99-11
32. Leontiadis I (2007) Publish/subscribe notification middleware for vehicular networks. In: Proceedings of the 4th international middleware doctoral symposium, Newport Beach, CA, USA, November
33. Chang YL, Hsu CC (2000) Routing in wireless/mobile ad hoc networks via dynamic group construction. Mob Networks Appl 5(1):27–37
34. Krishna P, Vaidya NH, Pradhan DK (1996) Static and dynamic location management in mobile wireless networks. Comput Commun (Special Issue on Mobile Computing) 19:321–334
35. Pagani E, Rossi GP (1999) Providing reliable and fault tolerant broadcast delivery in mobile ad hoc networks. Mob Networks Appl 4(3):175–192
36. Burns B, Brock O, Levine BN (2005) MV routing and capacity building in disruption tolerant networks. In: Proceedings of the annual IEEE conference on computer communications (INFOCOM), Miami, FL, USA, March

37. Lindgren A, Doria A, Schelèn O (2003) Probabilistic routing in intermittently connected networks. Mob Comput Commun Rev 7(3):19–20
38. Balasubramanian A, Levine B, Venkataramani A (2007) DTN routing as a resource allocation problem. In: Proceedings of the annual conference of the ACM special interest group on data communication(SIGCOMM), Kyoto, Japan, August
39. Widmer J, Le Boudec JY (2005) Network coding for efficient communication in extreme networks. In: Proceedings of the annual conference of the ACM special interest group on data communication (SIGCOMM), Philadelphia, PA, USA, August
40. Tchakountio F, Ramanathan R (2004) Anticipatory routing for highly mobile endpoints. In: Proceedings of the 6th IEEE workshop on mobile computing systems and applications, Lake District National Park, United Kingdom, December
41. Nain D, Petigara N, Balakrishnan H (2003) Integrated routing and storage for messaging applications in mobile ad hoc networks. In: Proceedings of modeling and optimization in mobile, ad hoc and wireless networks, INRIA Sophia-Antipolis, France, March
42. Mitra P, Poellabauer C (2012) Efficient group communications in location aware mobile ad-hoc networks. Pervasive Mob Comput (Elsevier) 8(2):229–248
43. Dubois-Ferriere H, Grossglauser M, Vetterli M (2003) Age matters: efficient route discovery in mobile ad hoc networks using encounter ages. In: Proceedings of the 4th ACM international symposium on mobile ad hoc networking and computing (MOBIHOC), Annapolis, MD, USA, June
44. Erramilli V, Chaintreau A, Crovella M, Diot C (2007) Diversity of forwarding paths in pocket switched networks. In: Proceedings of the 7th ACM SIGCOMM conference on internet measurement, San Diego, CA, USA, October
45. Hui P, Crowcroft J, Yoneki E (2008) 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), Hong Kong, China, May
46. Daly EM, Haahr M (2007) 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). Montreal, Canada, September
47. Mtibaa A, May M, Diot C, Ammar M (2010) PeopleRank: social opportunistic forwarding. In: Proceedings of IEEE INFOCOM mini conference, San Diego, CA, USA, March
48. Zyba G, Voelker GM, Ioannidis S, Diot C (2011) Dissemination in opportunistic mobile adhoc networks: the power of the crowd. In: Proceedings of IEEE INFOCOM mini conference, Shanghai, China, April
49. Juang P, Oki H, Wang Y (2002) Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet. ACM SIGPLAN Notices 37:96–107
50. Small T, Haas ZJ (2003) 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 of mobile ad hoc networking and computing, Annapolis, MD, USA, June
51. Pompili D, Melodia T, Akyildiz IF (2006) Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks. In: Proceedings of ACM annual international conference on mobile computing and networking (MOBICOM), Los Angeles, CA, USA, September
52. Pentland A, Fletcher R, Hasson A (2004) DakNet: rethinking connectivity in developing nations. IEEE Comput 37(1):78–83
53. Doria A, Uden M, Pandey DP (2002) Providing connectivity to the Saami nomadic community. In: Proceedings of the 2nd international conference on open collaborative design for sustainable innovation, Bangalore, India, December
54. Snekkenes E (2001) Concepts for personal location privacy policies. In: Proceedings of the 3rd ACM conference on electronic commerce, Tampa, FL, USA, October
55. Gorlach A, Heinemann A, Terpstra WW (2005) Survey on location privacy in pervasive computing. Privacy, security and trust within the context of pervasive computing. The Kluwer International Series in Engineering and Computer Science. Kluwer Academic, New York, pp 23–34

56. Kong J, Hong X (2003) ANODR: anonymous on demand routing with untraceable routes for mobile ad-hoc networks. In: Proceedings of the 4th ACM international symposium on mobile ad hoc networking and computing, Annapolis, MD, USA, June
57. Gruteser M, Grunwald D (2003) Anonymous usage of location-based services through spatial and temporal cloaking. In: Proceedings of the 1st international conference on mobile systems, applications, and services, San Francisco, CA, USA, May
58. Zhao W, Ammar M, Zegura E (2004) A message ferrying approach for data delivery in sparse mobile ad hoc networks. In: Proceedings of the 5th ACM international symposium on mobile ad hoc networking and computing, Tokyo, Japan, May
59. Chatzigiannakis I, Nikoletseas S, Spirakis P (2001) Analysis and experimental evaluation of an innovative and efficient routing protocol for ad-hoc mobile networks. Lecture Notes in Computer Science, vol 1982. Springer, Berlin
60. Jain S, Shah RC, Brunette W, Borriello G, Roy S (2006) Exploiting mobility for energy efficient data collection in wireless sensor networks. ACM/Kluwer Mob Networks Appl (MONET) 11(3):327–339
61. Ratsimor O, Finin T, Joshi A, Yesha Y (2003) eNcentive: a framework for intelligent marketing in mobile peer-to-peer environments. In: Proceedings of the 5th international conference on electronic commerce, Pittsburgh, PA, USA, September
62. Garyfalos A, Almeroth KC (2004) Coupon based incentive systems and the implications of equilibrium theory. In: Proceedings of the IEEE international conference on e-commerce technology, Taipei, Taiwan, March
63. Lilien L, Kamal ZH, Bhuse V, Gupta A (2006) Opportunistic networks: the concept and research challenges in privacy and security. In: Proceedings of the international workshop on research challenges in security and privacy for mobile and wireless networks (WSPWN), Miami, FL, USA, March

Chapter 7
Modeling of Intermittent Connectivity in Opportunistic Networks: The Case of Vehicular Ad hoc Networks

Anna Maria Vegni, Claudia Campolo, Antonella Molinaro and Thomas D. C. Little

Abstract This chapter analyses connectivity issues in a particular type of opportunistic networks: Vehicular Ad hoc NETworks (VANETs). The features of opportunistic networks well-fit VANETs, characterized by connectivity disruptions occurring due to quick network topology changes, high vehicle speed and variable vehicle densities. VANETs provide both intervehicle and vehicle-to-network-infrastructure communications. Vehicle-to-vehicle communications may not be the most appropriate interconnection scheme for data delivery in sparse or totally disconnected scenarios. Vehicle-to-infrastructure communications represent a viable solution to *bridge* the inherent network fragmentation that may exist in multi-hop networks formed over moving vehicles, but a ubiquitous roadside infrastructure can incur prohibitive deployment and maintenance costs. In this chapter, we present recent related work focusing on vehicular connectivity models and review hybrid and opportunistic vehicular communication paradigms designed to improve connectivity.

Keywords Opportunistic networking · Intermittent connectivity · VANETs

---

A. M. Vegni (✉)
Department of Engineering, University of Roma Tre, Rome, Italy
e-mail: amvegni@uniroma3.it

C. Campolo · A. Molinaro
Department DIMET, University Mediterranea of Reggio Calabria,
Reggio Calabria, Italy
e-mail: claudia.campolo@unirc.it

A. Molinaro
e-mail: antonella.molinaro@unirc.it

T. D. C. Little
Department of Electrical and Computer Engineering,
Boston University, Boston, MA, USA
e-mail: tdcl@bu.edu

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_7,

7.1 Introduction

Opportunistic networks are one of the most interesting evolutions of mobile ad hoc networks (MANETs). In opportunistic networks, the assumption of a complete path between the source and the destination is relaxed: mobile nodes are enabled to communicate with each other even if a route connecting them may not exist or may break frequently [1]. Traditional routing protocols for the Internet and MANETs [2] assume that an *end-to-end* path exists, thus they fail in opportunistic networks.

Opportunistic networking techniques allow mobile nodes to exchange messages by taking advantage of mobility and leveraging the *store-carry-and-forward* approach. According to this technique, a message can be stored in a node and forwarded over a wireless link as soon as a connection *opportunity* arises with a neighbor node, which is opportunistically used as a nexthop toward the destination [1].

Messages that are cached in the network and wait for an end-to-end path to be available, can suffer from additional delays in delivery. This is why opportunistic networks are also considered as a special kind of *delay tolerant network* (DTN) [3], providing connectivity despite long link delays or frequent link breaks that can be caused by nodes moving out of range, environmental changes, interference from other objects, etc. [4].

Opportunistic networks include: mobile sensor networks [5], pocket-switched networks [6], and vehicular ad hoc networks (VANETs) [7]. A VANET is a special kind of MANET in which packets are exchanged between mobile nodes (vehicles) traveling on constrained paths, in case of vehicle-to-vehicle (V2V) communications, and between vehicles and road-side access points (APs) ($a.k.a.$ road-side units, RSUs), in case of vehicle-to-infrastructure (V2I) communications [8].

There are several characteristics that differentiate VANETs from the MANETs. MANET applications are identical (or similar) to those enabled by the Internet. In contrast, although VANETs can support them, e.g., enhanced informative services, audio/video streaming, and generalized entertainment, they have been mainly designed to improve the quality of transportation through time-critical *safety* and *traffic management* applications [9]. Additionally, vehicular applications demand strict communications performance (e.g., timely and reliable message delivery) that are not always needed in conventional wireless networks.

Vehicles move with higher speeds as compared to MANET nodes: in MANETs speed ranges from 0 to 5 m/s, while in VANETs speed ranges from 0 to 40 m/s. Moreover, vehicular networks have to cope with variable network densities induced by traffic conditions and by mobility patterns: vehicles do not move independently of each other, but according to well-established traffic models where mobility is constrained by road topology, speed limits, and traffic lights.

All these unique features let VANETs well fit into the class of opportunistic networks. On the one hand, the highly *dynamic topologies* caused by different vehicle speeds and mobility patterns, e.g., vehicles traveling on the roadway in opposite directions, are characterized by frequent link breakages that strongly hinder stable and durable V2V communications. On the other hand, the *limited infrastructure* *coverage,* because of sparse RSUs settling, may cause *short-lived* and *intermittent* V2I connectivity.

In several research works, protocols and standards have been developed for supporting and improving short-range communications between vehicles and between vehicles and infrastructure. A special interest has been devoted to analyze *connectivity*, which has unique features in the vehicular environment and can be deeply influenced by factors such as vehicle density and speed, and radio communication range. Connectivity in VANETs has been studied through simulations [10–13, 23] and analytical evaluation [14–29].

The study of connectivity is not only important to evaluate the performance of VANETs and to understand packet exchange between vehicles and vehicles and RSUs, but its modeling and prediction are crucial in enabling network designers and providers to effectively improve network planning deployment and resource management, in order to meet applications' requirements.

The objective of this chapter is to provide a comprehensive understanding of connectivity in VANETs, one the most interesting instance of opportunistic networks. We provide an extended description of the main factors affecting connectivity and the models proposed in the literature to characterize it especially in intermittently connected vehicular networks, and we review hybrid communication paradigms designed to improve connectivity in challenging network conditions.

This chapter is organized as follows. In Sect. 7.2, we introduce vehicular networks, their envisioned set of applications and main features. In Sect. 7.3, we overview the main connectivity issues in VANETs. Recent models that analytically characterize V2V and V2I connectivity are surveyed in Sects. 7.4.1 and 7.4.2, respectively. In Sect. 7.5, we provide an overview of the main state-of-the-art representative solutions for improving connectivity performance in vehicular networks, particularly opportunistic approaches and hybrid V2I/V2V solutions. Finally, in Sect. 7.6 conclusive remarks will be summarized.

7.2 VANETs : An Overview

In the last few years, VANETs have interested several players, from automotive manufacturers and academia to governmental agencies and standardization bodies, mainly for the expected deep social and economical impact related to the wide variety of applications conceived for these environments. Vehicular applications are typically classified in (i) active road *safety* applications, (ii) *traffic efficiency* and *management* applications, and (iii) *comfort* and *infotainment* applications [9].

Active road safety-related applications are geared primarily toward avoiding the risk of car accidents and making safer driving by distributing information about hazards and obstacles. The basic idea is to broaden the range of perception of the driver beyond his/her field of vision and allowing him/her to react much quicker, thanks to alerts reception through wireless communications.

Transport efficiency and management applications focus on optimizing flows of vehicles by reducing travel time and avoiding traffic jam situations. These applications, such as enhanced route guidance/navigation, traffic light optimal scheduling, and lane merging assistance, while optimizing routes also allow reducing gas emissions and fuel consumption.

Although the primary purpose of VANETs is to enable safety applications, non-safety applications are expected to create commercial opportunities by increasing the number of vehicles equipped with *on-board* wireless devices, thus pushing *market penetration* of the technology and making it more cost-effective. Comfort and infotainment applications aim to provide the road traveller with information support and entertainment to make the journey more pleasant. They are so varied and range from traditional IP-based applications (e.g., media streaming, voice over IP, web browsing) to applications unique to the vehicular environment (e.g., point of interest advertisements, maps download, parking payments, automatic tolling services).

VANETs applications exhibit very heterogeneous requirements. The main concern for safety applications is finding reliable, low-latency, and efficient methods for disseminating safety messages. In contrast, non-safety applications have very different communication requirements, from no special real-time requirements of traveler information support applications, to guaranteed quality-of-service (QoS) needs of multimedia and interactive entertainment applications.

There are several wireless access technologies that may be used for vehicular communications: such as IEEE 802.11 and wireless wide area network (WWAN) technologies, like long-term evolution (LTE) and worldwide interoperability for microwave access (WiMAX) [9]. In recent years, there was a wide consensus on the use of IEEE 802.11, which is a mature, high-bandwidth and low-cost technology with the capability to well fit the multi-hop, distributed, unstable and ad hoc nature of vehicular environments. To this aim, the IEEE 802.11*p* standard [30] has been recently published as an amendment to IEEE 802.11; it is intended to operate with the IEEE 1609 protocol suite [31] to provide the wireless access in vehicular environments (WAVE) protocol stack.

In VANETs, there are three primary models for interconnecting vehicles, based on (i) network infrastructure, (ii) inter-vehicle communications, and (iii) hybrid configuration:

1. The first architecture is an *infrastructure-based solution* in which vehicles connect to a centralized server or a backbone network such as the Internet, with the help of road-side infrastructure, e.g., cellular base stations, IEEE 802.11 APs, IEEE 802.11*p* RSUs. This approach is illustrated in Fig. 7.1. The infrastructure nodes have the role of managing the network and providing connectivity to the backbone, e.g., the Internet, or they could serve as gateways for vehicles to communicate with transportation authorities. In this configuration, the roundtrip delays for data dissemination are potentially high, which make such solution unsuitable for safety applications. Connectivity in this model is subject to availability of infrastructure and often such solutions are cost intensive. Although the deployment of a dedicated vehicular network infrastructure, e.g., relying on IEEE 802.11*p* road-side

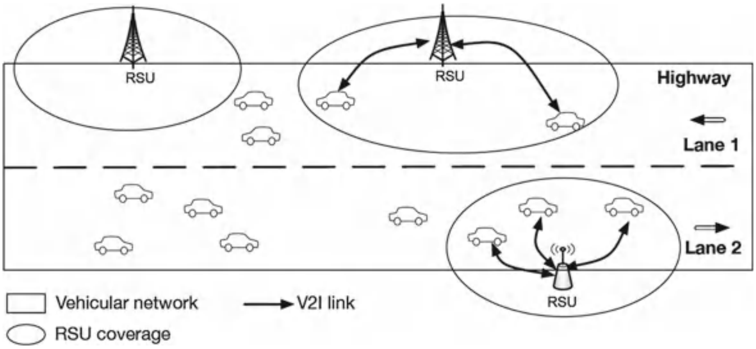


Fig. 7.1 An infrastructure-based solution for V2I communications. The RSU coverage is variable depending on the wireless technology

technology, cannot be considered as a short-term solution, IEEE 802.11 APs are widely available, especially in urban scenarios. Their low-cost, high-capacity, and low-coverage nature yields the opportunistic access to road-side communication infrastructures from traveling vehicles, known as *drive-through* [32];

2. The second solution exploits *direct ad hoc connectivity* among vehicles, as depicted in Fig. 7.2. Depending on the scope of the applications and on the network topology, packets may need to be forwarded at intermediate hops or not, therefore V2V can be either deployed as multi-hop or single-hop communications. For instance, multi-hop systems are useful for applications requiring long-range communications (e.g., traffic monitoring), while single-hop systems are indicated for applications requiring short-range communications (e.g., lane merging);
3. The third architectural solution is a *hybrid configuration* that proposes to use a combination of the two previous communication modes (i.e., V2V and V2I). Vehicles in range directly connect to the road-side infrastructure, assumed to be intermittently available, while they exploit multi-hop connectivity otherwise. The two communication protocols, V2V and V2I, have been developed as part of the *vehicle infrastructure integration (VII)* initiative [33], to be adopted in the vehicular environment. A conceptual VII system is shown in Fig. 7.3, where data information can be exchanged among vehicles, traffic management centres, as well as multimedia service providers for entertainment applications. Notice that the integration of V2V and V2I communications is in general called as inter-vehicle communication (IVC), that is, in an IVC system, vehicles and all road-side stations are assumed to have communication capabilities.

In summary, VANETs have some unique features, such as the constrained mobility pattern, the high vehicle mobility and the quickly changing network topology, ranging from sparse traffic densities to high car concentration in a small area, the multi-hop nature of V2V links, and the adverse effects of a hostile environment on the radio signal propagation. These unique features pose several issues to be addressed. Among them, modeling connectivity and designing solutions aiming at

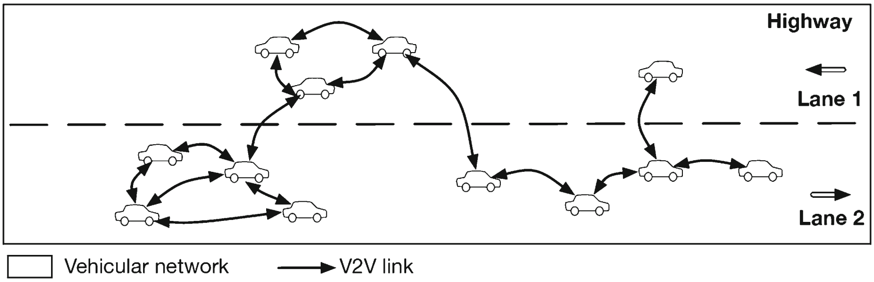


Fig. 7.2 Ad hoc connectivity via V2V

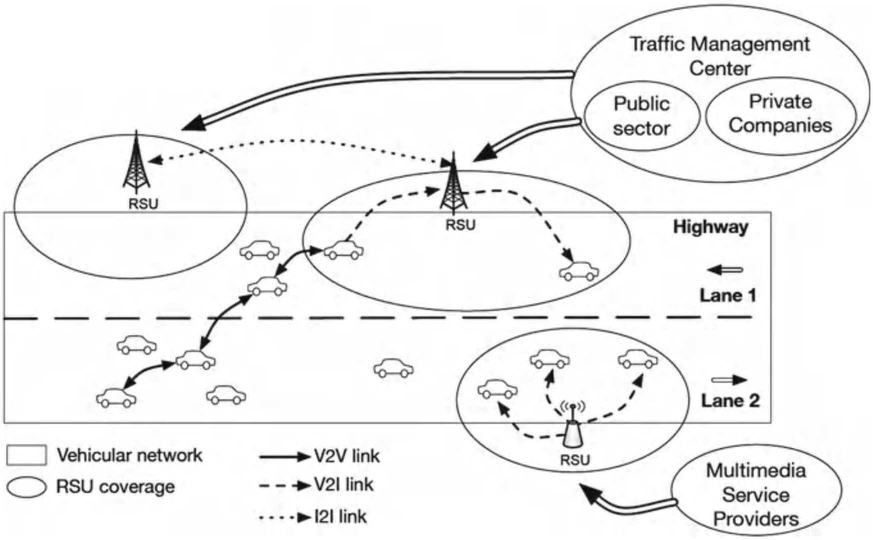


Fig. 7.3 Illustration of conceptual VII systems, as depicted in [29]

improving performance, by tolerating or reducing disruption when connectivity is sparse and by coping with congestion in crowded network conditions, represent the main challenging issues, as extensively discussed in next sections.

7.3 Connectivity in VANETs

Connectivity has a great impact on the performance of ad hoc networks, since it influences factors such as capacity, routing efficiency, and QoS of delivered applications, such as delay and reliability. This is why it has received a great interest from the research community.

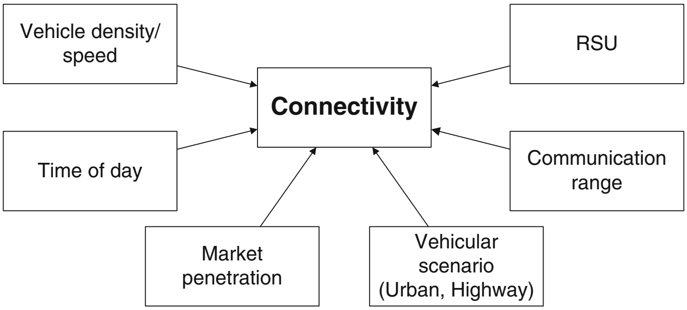


Fig. 7.4 Main factors affecting connectivity, as depicted in [14]

Although the connectivity analysis of VANETs is closely related to investigations on the connectivity of ad hoc and hybrid networks, many challenges and issues are peculiarities of the vehicular environment. Vehicles' connectivity is determined by a combination of several factors, as depicted in Fig. 7.4. Space and time dynamics of moving vehicles (i.e., vehicle density and speed), density of RSUs, and radio communication range strongly affect the connectivity in VANETs [14]. Most of these factors are mutually related.

Common mobility models for MANETs allow node mobility, determined by speed or pause time according to the random waypoint model (RWP) [34], to be considered independently of node density in performance evaluation. This is not the case for VANETs, where the average speed of vehicles is a function of vehicle density and vice versa (e.g., vehicles travel slower on congested roads). From the traffic theory [35] it is known that according to *macroscopic* vehicular traffic models, the behavior of each vehicle cannot be considered *separately*, as in *microscopic* models: all vehicles are aggregated into a flow which is described in terms of fundamental quantities including *vehicles density* (in vehicles per kilometer per lane), *flow* (in vehicles per hour per lane), and *speed* (in kilometers per hour). The values of these parameters are typically related by the so-called *fundamental traffic theory equation*, which is given in [35] by:

$$q = v \times k, \tag{7.1}$$

where q, v, and k are the traffic flow, average speed, and traffic density, respectively.

Over the years, many models have been proposed for speed–flow–density relationships [35, 36] but no single theory provides a complete picture of them.

Vehicle density varies significantly depending upon the time of the day and the location. Mornings and evenings are usually times of heavy traffic density—"rush hour"—and can suffer from congestion (*dense traffic condition*); while at off-peak times (e.g., night-time), roadways can be empty and intervehicle communications are not possible (*sparse traffic* and *totally-disconnected conditions*) [37]. Finally, urban areas yield high-density scenarios of slow moving traffic, while rural areas have relatively sparse population of vehicles. In urban environments, vehicle movements are restricted by the road topologies, buildings, etc., and affected by traffic density, which is determined by road capacity, traffic control, and driver behaviors. Traffic lights have also an important impact on connectivity: vehicles accumulate at red traffic lights in urban intersections by creating meeting points. However, as a drawback, a higher congestion can be experienced if the number of transmitting vehicles increases.

Vehicles traveling at different speeds are temporally able to communicate each other, and link disconnections can often occur. Things worsen if vehicles move in opposite directions, because the contact time interval gets shorter and limits the amount of data that vehicles can exchange.

Furthermore, the communication range has a direct impact on connectivity. Intuitively, a large transmission range must be chosen in order to keep the network connected, so that a vehicle can establish a link to any other vehicle in the network, either directly or over multiple hops. Typically, increasing the transmission range results in decreasing the number of hops between source and destination, leading to effectively increasing performance. A static transmission range cannot maintain the network connectivity due to the non-homogenous distribution of vehicles and the rapid traffic changes. However, in high-density conditions increasing the transmission range would cause severe degradation to the network performance because of high interference. Controlling the communication range by adjusting the transmission power can mitigate the adverse effects of high-density conditions, as proposed in [38].

It should be noticed that the actual transmission range depends on a number of factors, such as signal-to-interference-noise ratio (SINR) and receiver sensitivity. SINR on its turn depends on the transmission power, the interference, and the channel propagation features.

The *unit disk propagation* model [39] is largely used to describe channel propagation. It assumes that two mobile nodes are directly connected if and only if their Euclidean distance is less than or equal to the transmission range. Such a model is only suitable to model the radio environment on the highways, typically free of obstacles and buildings and characterized by line-of-sight (LOS) conditions [40] for short inter-vehicle distances. For larger distances, fading effects, mainly described by the Nakagami distribution [41], cannot be neglected causing serious additional and not deterministic attenuations. The log-normal shadowing model is used to model in a more realistic way signal propagation where the transmit power loss increases logarithmically with the Euclidean distance between two wireless nodes and varies log-normally due to the shadowing effect caused by surrounding environment [39].

Urban environments, conversely, are generally dominated by non-LOS (NLOS) communications, with multiple reflecting objects (e.g., buildings) that degrade the received signal quality and strength (*multipath effect*) [42, 43]. In both environments, due to the relatively low elevation of vehicle antennas, vehicles themselves (e.g. cars, trucks, buses) can act as obstacles to the signal propagation [44].

Intervehicle connectivity also depends on the *market penetration*; it is expected to be low at the initial phase of VANET deployment thus hindering V2V communications. Connectivity performance varies based on the penetration rate; unequipped vehicles physically occupy space and alter the spatial distribution of vehicles and their mobility [14]. Connectivity improves as the market penetration increases, since it directly translates in an increasing probability of finding a neighbor that forward messages. However, a higher number of equipped vehicles may increase the offered load to the network, thus increasing congestion.

Under quick topology network changes, mainly caused by vehicle speed, and in sparse (i.e., low density or low market penetration) or totally disconnected scenarios, vehicles are not always able to communicate, and V2V may not be the most appropriate interconnection scheme for some applications, especially non-safety critical ones [45, 46].

A solution for longer-range vehicular connectivity should consider pre-existing network infrastructure to enable V2I communications from vehicle to RSUs, and vice versa. A pervasive roadside infrastructure would be critical to encourage the adoption of IVC by individual drivers or car companies, since it is expected to take years before having a market penetration rate that can support efficient V2V communications.

However, the coverage of RSUs may not be complete due to the high costs for planning, deploying, and maintaining a ubiquitous road-side infrastructure. Candidate places for RSU installations could be service areas on a highway, or points of interests and crossroads in a city that leverage the already existing infrastructure (i.e., traffic lights, junction box, etc.). V2I connectivity also depends on the specific wireless technology for the RSUs (i.e., WiMAX, IEEE 802.11p, etc.) that influences the transmission range and radio coverage.

Most of VANET research has focused on analyzing VANETs as *well-connected networks*, e.g., often with the purpose of studying the *broadcast storm* problem caused by frequent contention and collisions due to redundant transmissions from neighboring vehicles in dense network topologies [47]. In contrast, especially at the early VANET deployment stages, as also argued in [48], we expect that the sparse RSU settling and the low market penetration rate, coupled with vehicle mobility and harsh radio propagation conditions, will result into intermittent, poor, and short-lived V2V and V2I connectivity. In this context, the design of reliable and efficient routing protocols that can support highly diverse and mainly intermittently connected network topologies is a challenging research topic, which may require the exploitation of opportunistic techniques.

7.4 Modeling Connectivity in VANETs: Literature Solutions

In this section, we explore recent related works coping with *how to model connectivity* in VANETs. Existing analytical models in the literature differ for the assumptions about vehicles distribution and mobility, the communication type, the number of communication hops, and the considered connectivity metrics. But, all of them share the same objective to provide important insights into how the network and road parameters, such as road topology, traffic density, vehicle speed and transmission range, affect the network connectivity behavior.

This section is organized in two subsections that, respectively, introduce the main related works addressing the connectivity issue in VANETs, on the basis of V2V and V2I communication modes.

7.4.1 Modeling V2V Connectivity

Vehicular connectivity represents an open issue since it is not always supported, and messages can be lost or never received.

Given the logistic challenges and the high costs of deploying and testing actual equipments in vehicles, simulations [10–13], and analytical models [14–23] are the only feasible and cost-efficient tools for analyzing the fundamental properties of V2V connectivity. However, due to their (potentially) higher computational efficiency compared to simulation-based approaches, analytical models are typically preferred. Moreover, they achieve more general findings compared to simulations that are, instead, always dependent on the simulation scenario.

The most of existing literature in VANET focuses on modeling the V2V *connectivity probability*. A largely common assumption in connectivity models is that a vehicular network is *partitioned* into a number of *clusters* [14, 15]: vehicles within a partition communicate either directly or through multiple hops, but no direct connection exists among partitions. Such an observation is also supported by simulations studies [13, 23].

Agarwal and Little [15] characterize a vehicular network by fast-paced vehicles traveling on constrained paths (roadways) with potentially short-lived connectivity. A *constant speed model* is assumed: vehicles exhibit different speed values (i.e., from 30 to 120 km/h) that do not change over time. The connectivity in the VANET is modeled through a graph of the network, which changes at a faster rate compared to MANET models. The *connectivity graph* is introduced as a graph, whose vertices correspond to vehicles equipped with an on-board vehicular communication unit and whose paths are defined by the short-range radio links; the connectivity over time can be then analyzed with snapshots of the graph. The graph may be adapted to a specific environment. Notably, the *Freeway* model restricts the vehicles' movement on several bi-directional multi-lane freeways (i.e., highways), while the *Manhattan* model restricts the movements to urban grids (i.e., roads with junctions) [8].

A time-varying traffic density implies a scenario in which there is lack of end-to-end connectivity between vehicles on the highway, and the network becomes *partitioned*. The connectivity graph formed by vehicles can be described as a partition yielding multiple disconnected subnets (*clusters*), as illustrated in Fig. 7.5.

Importantly, the work in [28] characterizes the performance of messaging in a fragmented network under the assumption of delay tolerant networking. Assuming exponentially distributed intervehicle distances, it is shown that vehicles, modeled as point objects whose length is not factored, traveling in the same direction are likely to be disconnected and the probability that two consecutive vehicles are disconnected is given in [15] as:

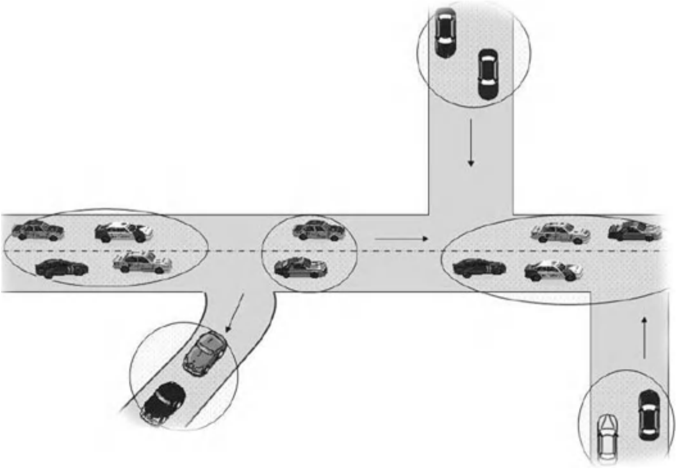


Fig. 7.5 Disconnected vehicle clusters, due to a gap among consecutive clusters, [16]

$$P\{X > R\} = e^{-\lambda R} \neq 0, \tag{7.2}$$

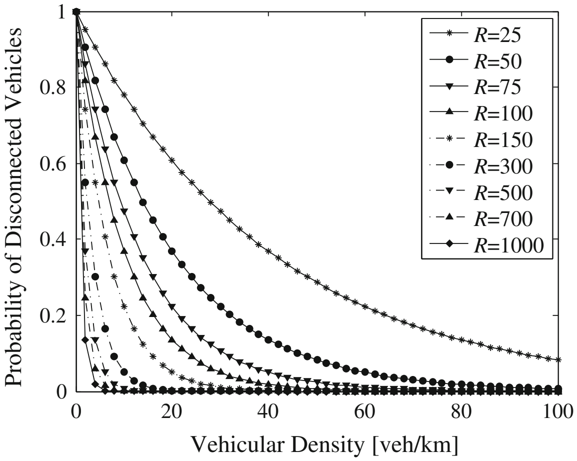
where X[m] is the intervehicle distance, λ [veh/m] is the distribution parameter for inter-vehicle distances, and R [m] is the radio range. Vehicles are considered to be connected if the intervehicle distance is less than the radio range, i.e., $X \leq R$. Correspondingly, vehicles are considered to be disconnected if the intervehicle distance is greater than the radio range, i.e., $X > R$.

The exponential distribution used to generate the intervehicle distances on the roadway has been shown to be in good agreement with real vehicular traces under uncongested traffic conditions [48]. The memoryless property of the exponential distribution [49] implies that the intervehicle distances are independent of each other. Figure 7.6 shows the connectivity trend as the density ranges from 0 to a 100 vehicles/km, for different values of the radio range parameter R [m]. We can notice that as the vehicle traffic density increases, the probability of disconnected vehicles decreases.

Similarly, for the same value of vehicle traffic, it is intuitive that vehicles are disconnected with increasing probability as the radio range decreases. It is significant to note that there is always a non-zero probability that vehicles are disconnected.

The work in [14] has analyzed the influence of a number of parameters on the V2V connectivity modeling, including (i) the vehicle density, (ii) the market penetration, and (iii) the radio communication range. The authors consider a hash-shaped grid of N vertical, and N horizontal roads; as in [28], they design the network through a *connectivity graph*. Two observed values are considered, i.e., (i) the fraction of vehicles not connected with any other vehicle, i.e., $\varphi(t)$, and (ii) the fraction of vehicles belonging to the largest connected component of the graph, i.e., $\theta(t)$. Vehicle clustering is addressed both analytically and through simulations. For scenarios with simple

Fig. 7.6 Disconnected vehicles probability versus the vehicle traffic density, for different communication range values



intersections, the authors show that accurate predictions of the network connectivity can be made using *percolation theory* [50], describing the behavior of connected clusters in a random graph.

In the stationary regime, the authors model the spatial distribution of vehicles with a *Poisson process*, and then compute an upper bound on the average fraction of vehicles that are connected to no other vehicles (i.e., $E[\varphi(t)]$). This represents the situation when the vehicular network is at a state that the rate of vehicles entering the network is the same as the rate of vehicle leaving it:

$$E[\varphi(t)] = e^{-2\lambda\rho R}, \tag{7.3}$$

where R [m] is the connectivity range enabling short-range intervehicle communications, and ρ [veh/m] is the vehicle density.

We remind that in one-dimensional networks, the knowledge of inter-node distances is necessary to analytically model connectivity in VANETs. Many authors rely on the assumption that the positions of the vehicles can be statistically modeled with a poisson point process (PPP), e.g., [14, 17, 18, 26, 27]. The PPP has two main properties, such as (i) the distance between two consecutive points is a random variable with an exponential distribution with parameter λ, and (ii) given $x \in \mathbb{R}^+$ the number of points falling in the finite interval $\mathcal{I} \overset{\Delta}{=} (0, x) \subset \mathbb{R}$ is a random variable with a Poisson distribution with parameter λx. Figure 7.7 shows an illustrative realization of a PPP with parameter λ. Denoting by n the number of Poisson points falling in, it is possible to define the n-dimensional positions vector as

$$\mathbb{R}^{(n)} = [R_1, R_2, \ldots, R_n], \tag{7.4}$$

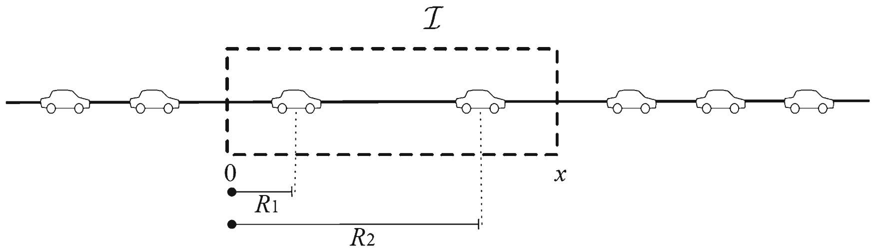


Fig. 7.7 An example of a vehicular linear network topology, where vehicles are placed following a Poisson process

where R_i, with $i \in \{1, 2, \ldots, n\}$, is the distance of the i-th point from the vehicle source placed in zero.

For what concerns the fraction of vehicles belonging to the largest connected component, i.e., $\theta(t)$, Kafsi et al. [14] consider the probability p that a road segment (i.e., the portion of road between two consecutive intersections) is covered by a sequence of connected vehicles. In this vision, the vehicular network is assumed as an *edge percolation model*, where each road segment is covered with probability p.

The use of percolation model allows addressing the case of a network with infinite size, with $N \to \infty$. According to [50] it is known that:

$$E[\theta(t)] = \begin{cases} 0 & \text{if } p < \frac{1}{2} \\ > 0 & \text{if } p < \frac{1}{2} \end{cases} \tag{7.5}$$

which is termed the *percolation phenomenon*. It means that there is a *critical density* above which a giant connected component appears (super-critical phase), while below this density, all connected components have a finite size. If the network is large but finite, its connectivity behavior is not much different. *Below* the critical density, all connected components are relatively small, and the largest of them may only contain a small fraction of the vehicles. On the contrary, *above* the threshold, a large connected component forms, containing typically more than half of the vehicles.

Therefore, a "soft" transition from below to above the critical density is expected, with the fraction of vehicles in the largest cluster suddenly shifts from low values to values close to one. Notice that these results are valid only if each road segment is covered independently with the probability p. It follows that the location of vehicles near intersections may influence the connectivity of adjacent segments. In addition, the authors show that traffic lights significantly influence the size of largest clusters, since the commonly observed accumulation of vehicles at red traffic lights can be beneficial for connectivity as it can sustain V2V communications. However, this clustering has the drawback of increasing the distance between equipped vehicles and its fluctuations. Thus, the largest cluster size remains low.

The presence of traffic lights is also considered in the analytical approach in [21]. The authors propose a stochastic traffic model, which relies on both the *fluid* model and *stochastic* model, in order to (i) characterize the general flow and the evolution of the traffic stream, and (ii) take into account the random behavior of individual vehicles. This method can be also utilized to characterize the connectivity dynamics inside the VANETs, modeled by stochastic traffic. With the knowledge of the vehicular density dynamics, the authors determine the probability that the network within an urban road segment is connected.

The use of percolation theory has been also used in [19] and [20]. Jin et al. in [19] give a theoretical analysis of the VANET connectivity in Manhattan grids by investigating the quantitative impact of (i) vehicle density, and (ii) transmission range on network connectivity. Jin et al. provide connectivity analysis in two different scenarios, featured with (i) small and (ii) large transmission ranges. Every road segment between two intersections can be regarded as a *bond* and the probability that a road segment is covered by a sequence of connected vehicles is denoted as p. A bond is said to be *open* with probability p, and then *closed* with probability $1 - p$. When the transmission range is small, it can be assumed that whether a bond is open or not is independent from other bonds, and the *bond percolation model* can be applied. This assumption does not hold for large transmission range scenarios for which the *Bollobás model* is better suited. In both cases, a network connectivity indicator *threshold* is derived, which determines whether the network connectivity is bad or not.

Given the vehicle density, the model allows calculating the minimum transmission range to achieve good network connectivity.

In [22] V2V connectivity is investigated in a highway scenario. Similar to some mentioned works, the authors assume the network can be partitioned and they focus on two metrics: (i) the platoon size (i.e., the number of vehicles in each connected cluster), and (ii) the connectivity distance (i.e., the length of a connected path from any vehicle). Unlike most of other related work that start from a given vehicle distribution on the road, by leaving unclear the effect of the vehicle's speed distribution and the road's traffic flow, the authors explicitly derive the distribution of the distances between cars. To more precisely study the effects of speed on connectivity, they provide bounds obtained by using stochastic ordering techniques, by following the work of Miorandi and Altman [51] for 1-D ad hoc networks, which transformed the problem of connectivity distance distribution into that of the distribution of the busy period of an equivalent infinite server queue. It was observed in [51] that the busy period of an infinite server queuing system has the same distribution as the connectivity distance. Moreover, the number of customers served during a busy period has the same distribution as the number of mobiles in a connected cluster in the ad hoc network.

The probability distribution and the expectations of the platoon size and the connectivity distance are both derived under different market penetration, number of lanes, transmission range, traffic flow, and speed scenarios. They found out that when the traffic's speed increases, the metrics of connectivity decrease. A further finding is that if the variance of the speed's distribution is increased, then, provided that the average speed remains fixed, the connectivity is improved.

Most of existing research studies focused on analyzing network connectivity in a simple highway scenario, modeled as 1D network topology, where vehicles travel in well-defined directions. Instead, a few studies exist that analyze connectivity in more complex two-dimensional urban scenario, [14, 21, 23]. The work in [23] provides a comprehensive framework for studying vehicular connectivity in urban VANETs by considering, in addition to [14, 21], the possible obstruction of signal propagation due to buildings. Besides extensive simulations based on their *Cellular Automata Model for Mobility* introduced in [52], the authors provide a statistical analysis of connectivity. The mobility model is designed to accurately describe traffic behavior in urban scenarios, with intersections, traffic lights, which exhibit great spatial diversity, making car distribution far from uniform.

Due to presence of intersections in urban scenarios, a link may be broken because of one or a combination of the following causes: (i) *non-intersection*, when two vehicles travel in directions away from each other causing the link to break when the distance increases beyond the transmission range; (ii) *light-indication*, when two vehicles traveling in the same direction sees red and green light, respectively; (iii) *turning-vehicle*, when the direction of vehicles change at the intersection. Closed form expressions of *link duration*, defined as the continuous time during which two vehicles are within transmission range of each other, are derived mainly through basic geometry, by assuming that: (i) vehicles move at *constant speed,* assumption supported by simulation results the authors achieved; (ii) vehicles have the *same transmission range;* and (iii) the *same probability* exists to *get a red or green light* upon arriving at intersections. The *re-healing* time, capturing the time duration in which there is no available paths between the two vehicles, is also derived starting from the assumption that it can be approximated by the time during which a vehicle is not in the largest connected cluster. Such an assumption is supported by simulation results showing that connectivity can be studied through cluster-based analysis, similarly to [14, 28], and that networks consist of one main cluster containing almost all of vehicles and other smaller clusters.

7.4.2 Modeling V2I Connectivity

Unlike V2V connectivity, difficult to be analyzed in large-scale test-beds, preliminary field-trials have been conducted to prove the *feasibility* of V2I communications to support Internet-based applications, e.g., [32, 53]. However, achieved performance are not promising: as a matter of fact, as vehicles move, their connectivity is both *fleeting*, usually lasting only a few seconds at urban speeds, and *intermittent*, with gaps between a connection and the subsequent one (i.e., vehicles could wait up to several minutes before obtaining again connectivity). The experimental study in [53] shows that the median vehicle-to-AP connection duration is around 13 s, while the mean duration between connections is around 75 s. Not negligible connectivity gaps between nearby Wi-Fi APs are also registered in experimental test-beds conducted in [24].

Only recently systematic approaches have been designed to model the (multi-hop) connectivity to infrastructure nodes [14, 24–29, 65]. In addition to the mobility and distribution patterns of vehicles, the market penetration rate of wireless technology in vehicles, and the transmission range, the distribution pattern of road-side stations would significantly affect the connectivity performance in V2I configurations.

Apart from addressing the connectivity in V2V communications, as previously described, Kafsi et al. in [14] study the influence of RSUs on vehicular connectivity. They assume that all RSUs are connected over wired or other fixed communication links, e.g., the Internet, in order to avoid intervehicle disconnections. The RSUs are placed at intersections, considering existing infrastructure at these locations (i.e., traffic lights, electricity, etc.), and have a distance of 400 m between each other.

The authors demonstrate that the differing vehicle placement conditions influence the overall connectivity and, most notably, the proportion of vehicles in the largest cluster. In contrast, RSUs do not significantly improve connectivity in all scenarios, e.g., RSUs at intersections do not reduce the proportion of isolated vehicles, which are more likely to be in the middle of the road, and then do not significantly increase the connectivity (i.e., the size of the largest cluster). As a consequence, RSU placements should be carefully planned and the authors suggest placing them in the middle of the road as a possible alternative. However, it should be noticed that results have been achieved with a *simplistic* channel model. Simulation results in [43] show, instead, that placing RSU at intersections can help to counteract corner propagation effects.

In [24] a Hidden Markov model (HMM) is proposed to model V2I connectivity. Single-hop V2I communications are considered, and vehicles experience connectivity *disruption* when they are not under AP coverage. In this context, the aim of the work is to model aggregate disruption in a particular drive. In fact, the HMM is based on experimental observations tracked through tests conducted by driving through existing domestic and commercial areas of a city where Wi-Fi connectivity is provided. Three states are considered in the HMM for modeling the connection state of vehicles: (i) *usable*, when a vehicle is connected to an AP and it is using its network services; (ii) *connected*, when a vehicle tries to get a connection with an AP; (iii) *disconnected*, when the vehicle is not under the coverage of an AP. Both open access APs (i.e., not requiring authentication) and closed access APs (i.e., requiring users to show proper credentials in order to access network services), are deployed in the considered scenario and accounted for in the proposed model. As a drawback, the main limitation of the work is that achieved results, e.g., in terms of the probability of encountering a given number of APs, are topology-specific and cannot generally represent all areas. However, this approach is valid for different areas, whenever the HMM state transition matrices are properly modified.

Understanding connectivity dynamics could have the additional benefit to help network designers and providers to effectively optimize the deployment of RSUs, that is a challenging task especially in case of disrupted V2I communications. In [25] it is introduced the notion of *intermittent coverage* for mobile users, called α-*coverage*, which provides the worst-case guarantees on the interconnection gap, while using significantly fewer RSUs. The *interconnection gap* is defined as the maximum distance, or expected travel time, between two consecutive vehicle-RSU contacts. Such a metric is chosen because the delay due to mobility and disconnection affects messages delivery more than channel congestion. As an analogy, the *crossing time* parameter defines the time a vehicle spends inside a wireless network, and represents a criterion for *handover* procedures in VANETs [54]. In [25] the authors rely on the *connectivity graph* concept, where RSUs are the vertex, and a weight is associated to each edge, i.e., a road segment that models the road length and the expected travel time. Informally, a deployment of RSUs provides α*-coverage*, if any simple path of length α in the network meets with at least one RSU. For a given deployment and a given budget on the number of RSUs, the proposed approach looks for an optimal RSU deployment providing α*-coverage* for a minimum possible α under the budget constraint. This problem is NP-hard, hence the authors propose efficient heuristics. The proposed framework can be used to decide how to place RSUs at roadside either from scratch or incrementally so that worst-case service guarantee can be given, e.g., for applications tolerating intermittent connectivity.

In [26] an analytical model is derived to fully characterize the connectivity in a vehicular network considering both one-hop (direct access) and two-hop (via a relay vehicle) communications between a vehicle and the infrastructure. Closed-form equations have been derived for calculating (i) the *access probability*, i.e., the probability that *any arbitrary vehicle* can access its nearby RSUs within two hops, for user satisfaction analysis, and (ii) the *connectivity probability*, i.e., the probability that *all vehicles* can access at least one RSU within two hops, by investigating a sub-network bounded by two adjacent base stations, for service coverage analysis.

RSUs are uniformly deployed along a road, while vehicles are distributed on the road randomly according to a Poisson distribution. Equations are derived for a generic radio channel model and specified for the unit disk and log-normal shadowing models. The impact of the following system parameters has been considered: inter-RSU distance, vehicle density, and transmission range. Achieved results show that (i) when the vehicle density is low, a vehicle is either directly connected to an RSU or disconnected, i.e., cannot reach any RSU in at most two hops, while (ii) with an increase in vehicle density the *access probability* increases, since the probability for vehicles in the gap of the RSU transmission ranges to find a neighbor within the transmission range of an RSU, acting as a relay node, increases. These results can be useful for a network operator to design a network with a given level of access guarantee. However, the main weakness of this approach is that it neglects mobility aspects, expected to significantly affect performance.

Adbrabou and Zhuang in [27] propose an analytical model, which is useful to approximately estimate the minimum number of RSUs, required for covering a road segment. The focus is on *low-density* vehicular networks, where V2I communications can be extended through vehicles relaying packets. Instead of modeling connectivity-related parameters, the framework targets the impact of several parameters (including the vehicle speed) on *packet delay* performance, in case of disrupted connectivity.

As in most of the existing literature, the model assumes that vehicles are distributed as Poisson points. The authors introduce a new vehicle mobility model that is characterized by two random variables, (i) the vehicle speed (i.e., V [m/s], in the range $[v_L, v_H]$), and (ii) the time period (i.e., T[s]) during which a vehicle moves

Table 7.1 List of the main common assumptions in connectivity models for VANETs

Assumption	Assumption type	Study
Vehicle distribution	Poisson	[14, 15, 17, 18, 21, 26–28, 59]
Topology	1D w/o traffic lights/intersections	[15, 18, 22, 24–29, 59]
Underlying model	Connectivity graph	[14, 15, 19, 20, 25, 28, 65]
Propagation model	Unit disk model	[14, 15, 19, 21, 26, 28, 59, 65]
Distribution of RSUs	Uniform	[26, 28, 59]

with a constant speed and is exponentially distributed to model the driving behavior. Independent mobility assumption is valid since it is jointly considered with vehicles bypassing each other. Unlike results in [26], in [27] multi-hop connections are considered and then packet delivery delay is not influenced by variations in the vehicle density or transmission range, as long as these variations keep the vehicular network sparse.

Similarly in [28], the authors consider the problem of access point placement. Under the assumption of delay tolerant messaging, they consider varying vehicular traffic densities and various RSU separations, to achieve connectivity. Different design choices can be taken as a function of vehicular traffic density, physical radio characteristics, and vehicle speed. The authors have shown that a large RSU separation is possible in a hybrid vehicular networking environment, comprised of multi-hop communication over moving vehicles, supported by RSUs.

To summarize the main contributions of this section, the following Table 7.1 is reported. It enlists the main common assumptions to the connectivity models analyzed in Sects. 7.4.1 and 7.4.2.

7.5 Solutions Improving VANET Connectivity

The connectivity models surveyed in the previous section show that (i) V2V communications can be effectively established only for dense traffic scenarios, but they are very limited in low-density neighborhoods, and (ii) short-lived connectivity and disruptions can be experienced in V2I communications.

In order to extend vehicular connectivity support, several solutions have been proposed in the literature, mainly leveraging *opportunistic* approaches. As a matter, the opportunistic contacts in vehicular networks, both among vehicles and from vehicles to available RSUs, can be very well used to instantiate and sustain both safety and non-safety applications. In this section, we present different solutions improving connectivity in VANETs, based on multi-hop V2V and V2I opportunistic links. Moreover, a brief description of opportunistic networking via satellite communications will be also investigated.

Opportunistic forwarding is the main technique adopted in DTN [55], and also extended in VANETs to achieve connectivity between vehicles via V2V and to disseminate information [22, 28, 56]. In particular in [28], when assuming a bidirectional road, the clusters formed in one direction of the highway come in *intermittent* contact with clusters of vehicles traveling in the opposite direction. Such contacts can be opportunistically exploited as a *bridging* technique, linking the partitioning that exists between clusters traveling in the same direction of the roadway.

On the other hand, the exploitation of RSUs together with inter-vehicle communications represents a viable solution to extend connectivity support in those scenarios where vehicles are not able to directly communicate, particularly in some applications, to *bridge* the inherent network fragmentation that exists in any multi-hop network through expensive connectivity infrastructure [29, 53, 57–59]. The use of a vehicular grid together with an opportunistic infrastructure placed on the roads can be a good solution to guarantee *seamless connectivity* in dynamic vehicular scenarios [59–61], and *hybrid communication paradigms* for vehicular networking are used to limit intermittent connectivity. In such approaches, connectivity can be provided by both existing network infrastructure (e.g., APs, and RSUs) through a V2I protocol, and traditional V2V networking [59, 62–66]. This combination is commonly referred to as V2X.

The cooperation and coexistence of these two different methods (i.e., V2V and V2I) can assure a good connectivity in VANETs. In [62], the authors propose a Cooperative Infrastructure Delivery Protocol, which allows vehicles to gather information about encountered RSUs through direct communication with the network infrastructure, and subsequent exchange messages with neighboring vehicles via V2V. The authors show the effectiveness of their approach, although it is limited to the message exchange about the infrastructure discovery. In [63], Wedel et al. use V2X communications for an enhanced navigation system, which intelligently helps drivers to circumnavigate congested roads and avoid traffic roadblocks. Their contribution highlights advantages of V2X communication protocols for numerous safety applications. In [64] Jeongwook et al. analyze the performance of a general hybrid communication protocol, based on the IEEE 802.11p WAVE system. Finally, in [65] a hybrid communication paradigm achieving seamless connectivity in VANETs is presented. Moreover, in [66] to satisfy QoS requirements, QoSHVCP technique is presented, based on QoS priorization. Through the use of a load-balancing mechanism, the failure of achieving QoS requirements is considered, so that the message propagation delay of high priority packets decreases regardless of the network overload. Simulation results have indicated how a hybrid approach provides lower delays, while maintaining the level of user's performance.

Among many performance metrics of an IVC system is the multi-hop connectivity between two communication nodes: the connectivity between two equipped vehicles or RSUs is defined as the probability that a *multi-hop communication path* exists between them at a time instant [57, 59]. In [59], the authors have designed a hybrid communication protocol, simply named as *vehicle-to-X* (V2X), working in heterogeneous vehicular network scenarios, where overlapping wireless networks partially cover the vehicular grid. This approach relies on the concept of *multi-hop communication path*.

V2X takes origin from the main limitations of V2V and V2I communications, causing seamless vehicular connectivity management to be a challenge for VANETs. V2X leverages the advantages of both protocols, so that vehicles can *opportunistically* exploit the network infrastructure whenever inter-vehicle communications are not available. As a result, a vehicle in V2X can switch from V2V to V2I, and vice versa, on the basis of a decision algorithm that is executed in a distributed fashion by each vehicle, based on a cost function using path alternatives. It follows that V2X allows vehicles to "handover" from V2V to V2I, and vice versa. Each vehicle is assumed staying in a connectivity state whenever is connected via V2V or V2I. The handover from a serving communication protocol to the other one is initiated by each vehicle on the basis of a decision policy, namely *optimal path selection technique*, that chooses the optimal vehicular communication protocol between two end nodes (i.e., V2V or V2I).

This approach considers a *total cost function*, i.e., a linear combination of two physical parameters: (i) the radio resource utilization time, and (ii) the time interval needed to transmit a message over a *path*. An *optimal path* connecting the i-th vehicle to the k-th RSU via multi-hop is selected on the basis of a minimization process of the total cost function.

The hybrid vehicular scenario considered in [59] is depicted in Fig. 7.3, where (i) vehicles are traveling forward (reverse) in opposite directions, and (ii) several RSUs of different wireless technologies partially cover a given area. A vehicle is said to be disconnected from the forward (reverse) network if it is not connected to any forward (reverse) neighbors in the network, assuming that the vehicle moves forward (reverse). Moreover, since it is a bi-dimensional vehicular grid, and the *bridging approach* occurs, a vehicle is said to be disconnected from the forward (reverse) network if it is not connected to any forward (reverse) neighbors driving in the opposite direction. In other words, the network is disconnected if the separation between any two adjacent and opposite nodes is greater than a connectivity bond. However, this definition is valid for a whole network, while for a portion of network, it is said to be connected if, for every pair of end-vehicles, there exists a *path* between them; otherwise, it is disconnected.

Simulation results in [59] have shown that both in sparse and dense traffic scenarios, the optimum path can guarantee minimum total cost functions for vehicles connected via V2V, while high values are obtained with V2I in a dense traffic scenario, and with V2V in a sparse traffic scenario for increasing number of hops. The authors found out that V2V is most suitable in dense scenarios, and in sparse traffic neighborhoods for path with a limited number of hops; while V2I could be the most appropriate protocol in sparse scenarios when the number of hops linking a source to a destination is increasing. By alternating V2V and V2I opportunistically, seamless vehicular connectivity can be guaranteed. The potentiality of hybrid communication protocols benefits not only connectivity in disconnected scenarios, but also improves the message dissemination in VANETs. In [65] and [66] the authors rely on V2X approach and consider the *vehicular partitioning*, such as the vehicular network is

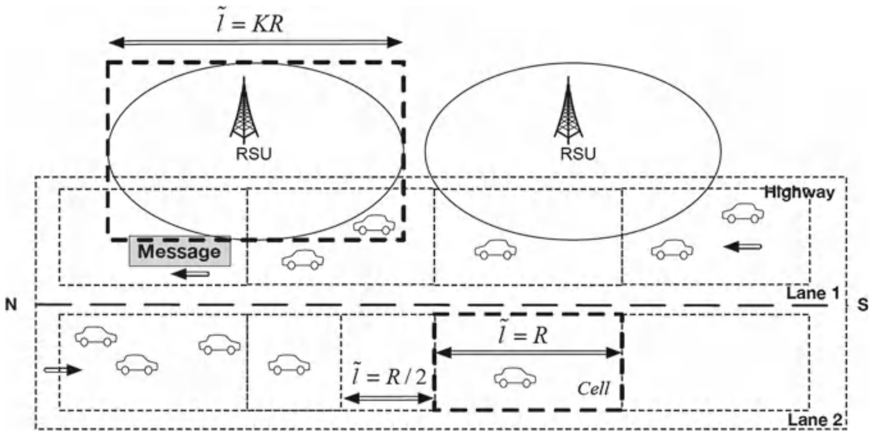


Fig. 7.8 Vehicular grid comprised of wireless RSU and V2V cells, [66]

comprised of different connectivity phases. The overall system can be modeled as an *alternating renewal process* where the vehicular connectivity structure alternates between three phases as follows:

1. Phase 1 (*No connectivity*): A vehicle is traveling alone in the vehicular grid. It represents a typical totally-disconnected traffic scenario where neither V2V nor V2I connectivity is available. The vehicles are completely disconnected;
2. Phase 2 (*Short-range connectivity*): A vehicle is traveling in the vehicular grid and forming a cluster with other vehicles. Only V2V connectivity is available within the transmission range of the sender/forwarder. No V2I connectivity is assumed to be available;
3. Phase 3 (*Long-range connectivity*): A vehicle is traveling in the vehicular grid with available neighboring RSUs. The vehicles enter the RSU coverage and are forced to connect via V2I to the Internet with accessible network infrastructure. No V2V connectivity is assumed to be available.

The probability that a vehicle lays in one of the three phases can be expressed as the probability that a vehicle is not connected, connected with neighbors, and with RSUs, respectively. In order to determine such a probability, it is useful to assume that a vehicular grid is discretized in terms of a number of cells, that is, the gap between two vehicles is equivalent to N cells. Figure 7.8 depicts the vehicular grid as composed of virtual RSUs and wireless V2V cells, with a variable size (i.e., $\tilde{l}$ [m]).

A cell is occupied if one or more vehicles are positioned within that cell. For a vehicle traveling alone on the southbound, i.e., S, (northbound, i.e., N), the probability that it will be connected in Phase 1 via multi-hop with a next vehicle on the southbound (northbound) depends on whether each of the N southbound (northbound) cells within the gap is occupied by at least one vehicle, given in [66] as

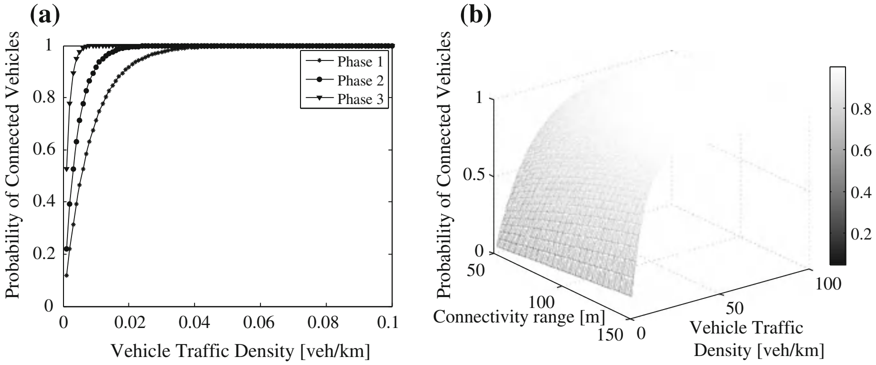


Fig. 7.9 Probability of connected vehicles (**a**) versus the vehicle traffic density (Phases 1–3), and (**b**) versus the vehicle traffic density and the connectivity range (Phase 1)

$$(p_{s,n})^N = (1 - \exp(-\lambda_{s,n} R))^N, \tag{7.6}$$

where $\lambda_{s,n}$ [veh/km] is the traffic density distribution on southbound and northbound, respectively. In this case, the number of cell is $N = 1$ since the gap equals the minimum intervehicle distance, i.e., $G = R$ [m].

In Phase 2, the vehicles along southbound (northbound) are connected via V2V if each of the N northbound (southbound) cells in the gap is occupied by at least one vehicle. This event occurs with probability $(p_{s,n})^N$, with $\lfloor N = G/R \rfloor$. In the event that not all of the N cells in the northbound direction are occupied, the vehicles along southbound are deemed to be disconnected.

Finally, in Phase 3, the probability that a vehicle traveling in the northbound (southbound) will be connected via V2I with a northbound (southbound) next vehicle depends on if each of the N northbound (southbound) cells in the gap is occupied by at least one RSU. In this case, $N = \lfloor G/(KR) \rfloor$ since the RSU cells are assumed to have a cell size $\tilde{l} = KR$ [m], with $K > 0$.

Figure 7.9a shows the analytical trend of the probability of connected vehicles for three different connectivity phases. The probability has been evaluated for R= 125 m; in Phase 2 vehicles are assumed to be separated for $G = 150$ m, while in Phase 3 the RSU wireless cells are greater than V2V cells for $K = 1.5$, and vehicles are separated for $G = 1$ km. Finally, in Fig. 7.9b the probability of connected vehicles in Phase 1 is shown. As expected, it depends on the availability of opportunistic links: the higher the vehicular density, as well as the connectivity range, the higher the probability of connected vehicles. The same trend can be obtained for connectivity in Phases 2 and 3.

In [67] the authors suggest to exploit RSUs to route packets between any source and destination in the VANET. RSUs, typically connected through a fixed infrastructure, act as a backbone and can communicate among each other to connect far vehicles. More in detail, when a vehicle has to send a packet to a remote vehicle it can send it to the nearest RSU, which in turn, forwards the packet to the nearest RSU to the destination. The benefit of such a solution is that it copes with the issue of finding, keeping, and updating a routing path to a specific destination, which could be difficult, if not impossible, in some vehicular environments because of fast topological changes and network fragmentation.

In previously surveyed works, we described how V2I communications can be opportunistically exploited to improve V2V connectivity, but solutions have been also proposed that leverage V2V support as a complement to possible intermittent connectivity with the road infrastructure, as mentioned in Sect. 7.4.2. The use of relay-based techniques have been proposed in [68–70], they lengthen the connection time with an AP and result in an increase in throughput.

In [68] a relay-based solution is proposed to extend the service range of roadside APs. As a vehicle moves towards an AP, its signal quality with the AP may be poor. A vehicle interested in uploading data towards the AP selects both a vehicle geographically ahead of it to serve as a relay, as well as a vehicle behind it to serve as a relay when it leaves the AP coverage area. Similarly, Yoo et al. in [69] leverage V2V relay support as a complement to possible intermittent connectivity with the road infrastructure, for downlink applications. More in detail, if an AP has a data frame destined to a vehicle that is not within its range, it sends the frame to another vehicle closer to the destination vehicle, acting as a *relay*. The relay then will deliver this frame to the destination vehicle. A solution improving V2I connectivity through vehicular relay is also proposed in [70]. Unlike previous works, relying on *drive-thru* connectivity offered by already existing Wi-Fi networks, the proposal has been designed to be compliant with upcoming IEEE 802.11p WAVE standards.

All previous approaches rely on V2V and V2I paradigms, for short- and long-range communications, respectively. However, in totally disconnected scenarios neither V2V nor V2I approaches are available. In this situation, when safety-critical messages need to be transferred, vehicular connectivity could be achieved *only* by satellite communication links, as investigated in [71]. Satellite radio is one of a complementary set of network connectivity technologies in future vehicles equipped with *on-board* computers. Main strength points of satellite links are a global coverage and the availability of broadcast and multicast services; however, link budget analysis is challenging, as well as the size and form factor of *on-board* antennas, in some cases, are unacceptable compared to terrestrial solutions.

Satellite connectivity has been largely used in VANETs for outdoor navigation and positioning services, while search-and-rescue (SAR) applications are used mainly in emergency scenarios to provide search for and aid to people in distress [8]. The scheme proposed in [71], as depicted in Fig. 7.10, shows how the satellite connectivity can solve the problem of seamless and ubiquitous connectivity, when a vehicle is driving alone in an area that is devoid of network infrastructure (e.g., a rural area during night hours), or it is in a disaster and emergency situation. Such vehicle is sometimes called as *isolated vehicle* (or vehicle in distress) in a totally-disconnected area. Satellite connectivity, adopted in VANET as an opportunistic link, is intended to augment short and medium-range communications to *bridge* isolated vehicles or clusters of vehicles when no other mechanism is available. Particularly, an isolated vehicle needs to alert about an accident occurred; it is not able to communicate to any

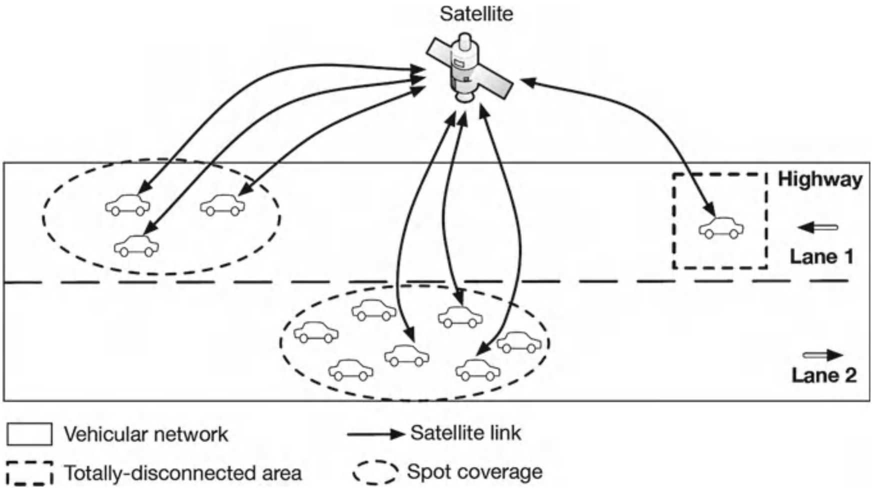


Fig. 7.10 An opportunistic networking scheme in a VANET scenario with satellite connectivity for safety applications. The satellite bridges an isolated vehicle, [71]

neighboring vehicles neither by V2V or V2I, but it can only send an SOS message via satellite links. An SOS message stores the vehicle's position. When the satellite receives the message, it will forward to the ground by *spot coverage* to clusters of vehicles in proximity. In the return link an acknowledgment is sent to the satellite in visibility, which forwards it to the isolated vehicle. The satellite works as a *bridge*, in order to connect the vehicle to the closest cluster of vehicles.

In [71] a dedicated low earth orbit (LEO)/medium earth orbit (MEO) Ka-band link analysis has been carried out, in order to evaluate (i) the minimum distance among cluster of vehicles and isolated user versus satellite orbit (i.e., LEO/MEO trade-off), (ii) the service availability along a selected time window, (iii) the LEO/MEO satellites visibility from uplink and downlink coverage, (iv) link feasibility and availability. The exploitation of satellite links in VANETs may provide a reduction of terrestrial network infrastructure, as well as multi-hop V2V communications, and a service coverage extension with respect to terrestrial infrastructure. Although promising, such a solution could be too expensive and requires further investigations.

7.6 Conclusions and Discussion

In this chapter, we have investigated connectivity issues in VANETs, i.e., a particular class of *opportunistic networks* in which, due to the dynamic behavior of vehicles and sparse RSUs settling, V2V and V2I connectivity links are short-lived and intermittent and have to be opportunistically exploited in order to transmit/forward messages.

We have described the main network and road parameters, such as road topology, traffic density, vehicle speed, market penetration of the VANET technology, and transmission range, which strongly affect the network connectivity behavior.

Existing analytical models in the literature have been investigated; each of them differs into the underlying assumptions and the considered connectivity metrics. However, some common assumptions have been noticed, as summarized in Table 7.1. For instance, most of the approaches rely on the Poisson model to describe vehicle distribution. Notice that, although the model is commonly used and widely accepted also in transportation engineering [35, 72], it does not capture the realistic interaction between vehicles, as outlined in [18]. First, positions of vehicles are not independent since drivers have to observe car-following or lane-changing rules. Second, the density of vehicles can vary significantly along a roadway due to both driving behavior and restrictions of network geometry.

Mobility dynamics are often completely neglected in existing studies thus hindering the possibility to effectively evaluate the performance of session-based applications. Studies evaluating connectivity metrics but neglecting connectivity dynamics, i.e., how connectivity between two vehicles changes with respect to time, are meaningful only when considering the dissemination of safety messages, which typically foresees the transmission of a single packet. However, session-based applications, like non-safety applications, may require the exchange of several packets, and in this case connectivity dynamics should be carefully considered.

With respect to connectivity metrics, besides the straightforward probability of connected (or disconnected) vehicles, other helpful metrics have been found in the literature. Link duration and crossing time, respectively, accounting for the connection duration among vehicles and between vehicles and infrastructure nodes, can determine whether a particular application can be supported in such an environment, e.g., Internet-based applications may not be feasible if contact times are below a given value. From a protocol designer's perspective, metrics like the infrastructure interconnection gap and the re-healing time may be used, respectively, to plan RSU placements and to indicate the size of a message buffer in case of *store-carry-and-forward* approaches carried out in V2V configurations.

It should be further noticed that, although not properly considered in several existing work, potential resource contention among several vehicles, concurrently accessing the medium to exchange data with nearby vehicles or RSUs, may significantly affect the quality and time duration of contacts.

Several works rely on the simplistic unit disk model to characterize channel phenomena, while realistic propagation models, accounting for shadowing effects and obstacles in urban scenarios, are considered only in a few works. Finally, another assumption simplifying the analytical treatment considers traditional 1D network topologies. It is the authors' convincement that since analytical models are expected to play an important role in performance evaluation of VANETs, they need to be significantly improved in terms of accurateness and realism.

Different solutions improving connectivity in VANETs have been presented. Some solutions exploit the presence of infrastructure nodes either to facilitate routing between distant vehicles or to bridge fragmented networks. Other approaches leverage handover mechanisms between traditional multi-hop V2V and V2I paradigms. Opportunistic networking via satellite communications is proposed to bridge isolated vehicles when no other mechanism is available. On the other hand, V2V relay-based techniques appear as a promising solution to "virtually" extend the infrastructure coverage and reduce connectivity disruptions. Further efforts are then required to design solutions enabling V2V and V2I connectivity in different network conditions to sustain both safety and non-safety applications.

References

1. Pelusi L, Passerella A, Conti A (2006) Opportunistic networking: data forwarding in disconnected mobile ad Hoc networks IEEE Commun Mag 44(11):134–141
2. Perkins CE, Belding-Royer EM, Das S (2003) Ad hoc on demand distance vector (AODV) routing, IETF, RFC 3561
3. Fall K (2003) A delay-tolerant network architecture for challenged internets. In: Proceedings of special interest group on data communications, Karlsruhe, Germany, pp 27–34, Aug 2003
4. Song L, Kotz DF (2007) Evaluating opportunistic routing protocols with large realistic contact traces, In: Proceedings of CHANTS '07, Montréal, Québec, Canada, Sept 2007
5. Eisenman SB, Lane ND, Campbell AT (2008) Techniques for improving opportunistic sensor networking performance. In: Proceedings of the 4th IEEE international conference on distributed computing in sensor systems, pp 157–175
6. Hui P et al (2005) Pocket switched networks and human mobility in conference environments. In: Proceedings of ACM SIGCOMM workshop on delay tolerant networking, pp 244–251, Aug 2005
7. Leontiadis I, Mascolo C (2007) Opportunistic spatio-temporal dissemination system for vehicular networks. In: Proceedings of MobiOpp 2007
8. Harteinstein H, Labertaux KP (eds) (2010) VANET vehicular applications and inter-networking technologies. Wiley, Chichester
9. Papadimitratos P et al (Nov. 2009) Vehicular communication systems: enabling technologies, applications and future outlook on intelligent transportation. IEEE Commun Mag 47(11):84–95
10. Fiore M, Harri J (2008) The networking shape of vehicular mobility. In: Proceedings of ACM international symposium on mobile ad hoc networking and computing, pp 261–272
11. Boban M, Misek G, Tonguz OK (2008) What is the best achievable QoS for unicast routing in VANETs. In: Proceedings of IEEE global communications conference, pp 1–10
12. Conceicao H, Ferreira M, Barros J (2009) A cautionary view of mobility and connectivity modeling in vehicular ad-hoc networks. In: Proceedings of IEEE vehicular technology conference, pp 1–5
13. Wu H et al (2005) Simulated vehicle-to-vehicle message propagation efficiency on Atlanta's I75 corridor. In: Proceedings of transportation research board conference, Washington, D.C.
14. Kafsi M et al (2008) VANET connectivity analysis. In: Proceedings of IEEE workshop on automotive networking and applications
15. Agarwal A, Little TDC (2010) Opportunistic networking in delay tolerant vehicular ad hoc networks. In: Watfa M (ed) Advances in vehicular ad-hoc networks: developments and challenges, pp 282–300. IGI Global, Hershey
16. Vegni AM, Stramacci A, Natalizio E (2012) SRB: a selective reliable broadcast protocol for safety applications in VANET. In: Proceedings of international conference on selected topics in mobile & wireless networking (iCOST, 2012), Avignon, France, July 2–4
17. El-Atty SMA, Stamatiou GK (2010) Performance analysis of Multihop Connectivity in VANET. In: Proceedings of 7th international symposium on wireless communication systems, pp 335–339, Sept 2010

18. Jin WL, Recker WR (Jan. 2010) An analytical model of multihop connectivity of inter-vehicle communication systems. IEEE Trans Veh Technol 9(1):106–112
19. Jin X, Su W, Wei Y (2011) Quantitative analysis of the VANET connectivity: theory and application. In: IEEE 73rd vehicular technology conference (VTC Spring), pp 1–5, May 2011
20. Mylonas Y, Pitsillides A, Lestas M (2008) Speed adaptive probabilistic flooding in VANETs. In: Proceedings of international trade and freight transportation conference, pp 66–73
21. Ho IWH, Leung KK, Polak JW (Feb. 2011) Stochastic model and connectivity dynamics for VANETs in signalized road systems. IEEE/ACM Trans Networking 19(1):195–208
22. Yousefi S, Altman E, El-Azouzi R, Fathy M (2008) Analytical model for connectivity in vehicular ad hoc networks. IEEE Trans Veh Technol 57(6):3341–3356
23. Viriyasitavat W, Bai F, Tonguz OK (Jan. 2011) Dynamics of network connectivity in urban vehicular networks. IEEE J Sel Areas Commun 29(1):515–533
24. Hasan SF, Ding X, Siddique NH, Chakraborty S (Jan. 2011) Measuring disruption in vehicular communications. IEEE Trans Veh Technol 60(1):148–159
25. Zheng Z, Sinha P, Kumar S (2012) Sparse Wi-Fi deployment for vehicular internet access with bounded interconnection gap. IEEE/ACM Trans Networking 20:956–969
26. Ng SC et al (Jan. 2011) Analysis of access and connectivity probabilities in vehicular relay networks. IEEE J Sel Areas Commun 29(1):140–150
27. Abdrabou A, Zhuang W (Jan. 2011) Probabilistic delay control and road side unit placement for vehicular ad hoc networks with disrupted connectivity. IEEE J Sel Areas Commun 29(1):129–139
28. Agarwal A, Little TDC (2008) Access point placement in vehicular networking. In: Proceedings of 1st international conference on wireless access in vehicular environments, Troy, MI, Dec 2008
29. Jin W-L, Wang H-J (2008) Modeling connectivity of inter-vehicle communication systems with road-side stations. Open Transp J 2:1–6
30. IEEE Std. 802.11p (2010) Wireless access in vehicular environments, July 2010
31. Uzcategui R, Acosta-Marum G (2009) WAVE: a tutorial. IEEE Commun Mag 47(5):126–133
32. Ott J, Kutscher D (2004) Drive-thru internet: IEEE 802.11b for automobile users. In: Proceedings of annual joint conference of the IEEE computer and communications societies, INFOCOM, Hong Kong, Mar 2004
33. Dong X et al (2006) Expediting vehicle infrastructure integration (EVII). Technical report, California PATH Research Report
34. Broch J et al (1998) A performance comparison of multi-hop wireless ad hoc network routing protocols. In: Proceedings of ACM MobiCom, pp 85–97
35. Roess RP, Prassas ES, McShane WR (2004) Traffic engineering, 3rd edn. Prentice Hall, Englewood Cliffs
36. Fricker JD, Whitford RK (2004) Fundamentals of transportation engineering: a multimodal systems approach. Prentice Hall, Upper Saddle River
37. Tonguz O et al (2007) Broadcasting in VANET. In: Proceedings of mobile networking for vehicular environments, pp 7–12, Anchorage, AK
38. Artimy M, Robertson W, Phillips W (2004) Assignment of dynamic transmission range based on estimation of vehicle density. In: Proceedings of the 1st ACM international workshop on vehicular ad hoc networks, Philadelphia, Pennsylvania, Oct 2004
39. Rappaport TS (2001) Wireless communications: principles and practice, 2nd Edn. Prentice Hall, Englewood Cliffs
40. Zang Y et al (2005) An error model for inter-vehicle communications in highway scenarios at 5.9 GHz. In: Proceedings of ACM PE-WASUN, Montreal
41. Taliwal V et al (2004) Empirical determination of channel characteristics for DSRC vehicle-to-vehicle communication. In: Proceedings of ACM VANET
42. Giordano E, Frank R, Pau G, Gerla M (2010) CORNER: a realistic urban propagation model for VANET. In: Proceedings of WONS
43. Campolo C et al (2011) Vehicular connectivity in urban scenarios: effectiveness and potential of roadside, moving WAVE providers and hybrid solutions. EURASIP J Wirel Commun Networking 146. doi:10.1186/1687-1499-2011-146, Published October 28

44. Boban M et al (2011) Impact of vehicles as obstacles in vehicular ad hoc networks. IEEE J Sel Areas Commun 29(1):15–28
45. Chiara BD, Deflorio F, Diwan S (2009) Assessing the effects of inter-vehicle communication systems on road safety. IET Intell Transp Syst 3(2):225–235
46. Moustafa H, Zhang Y (eds) (2009) Vehicular networks: techniques. Standards and applications. Auerbach publishers, Taylor and Francis Group, USA
47. Wisitpongphan N et al (2007) Broadcast storm mitigation techniques in vehicular ad hoc wireless networks. IEEE Wirel Commun Mag 14:84–94
48. Wisitpongphan N, Bai F, Mudalige P, Tonguz OK (2007) On the routing problem in disconnected vehicular ad-hoc networks. In: Proceedings of IEEE international conference on computer communications (INFOCOM ’07), pp 2291–2295, May 2007
49. Ross SM (2004) Introduction to probability models. Academic Press, New York
50. Grimmett G (1989) Percolation, 1 edn. Springer-Verlag, New York
51. Miorandi D, Altman E (2006) Connectivity in one-dimensional ad hoc networks: a queuing theoretical approach. Wirel Netw 12(6):573–587
52. Tonguz OK, Viriyasitavat W, Bai F (2009) Modeling urban traffic: a cellular automata approach. IEEE Commun Mag 47(5):142–150
53. Bychkovsky V et al (2006) A measurement study of vehicular internet access using in situ Wi-Fi networks. In: Proceedings of ACM annual international conference on mobile computing and networking, Los Angeles, CA, USA, Sept 2006
54. Esposito F, Vegni AM, Matta I, Neri A (2010) On modeling speed-based vertical handovers in vehicular networks—dad, slow down, I am watching the movie—. In: Proceedings of IEEE Globecom workshop on seamless wireless mobility, Miami, FL, USA
55. Spyropoulos T, Psounis K, Raghavendra C (2005) Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In: Proceedings of ACM SIGCOMM workshop on delay-tolerant networking, PA, USA
56. Resta G, Santi P, Simon J (2007) Analysis of multi-hop emergency message propagation in vehicular ad hoc networks. In: Proceedings of the 8th ACM international symposium on mobile ad hoc networking and computing, Montreal, Canada, pp 140–149
57. Briesemeister L, Schafers L, Hommel G (2000) Disseminating messages among highly mobile hosts based on inter-vehicle communication. In: Proceedings of IEEE intelligent vehicles symposium, pp 522–27, Oct 2000
58. Jin W-L, Recker WW (Mar. 2006) Instantaneous information propagation in a traffic stream through inter-vehicle communication. Transp Res Part B Methodol 40(3):230–250
59. Vegni AM, Little TDC (2011) Hybrid vehicular communications based on V2V–V2I protocol switching. IJVICS 2(3/4):213–231
60. Gerla M et al (2006) Vehicular grid communications: the role of the internet infrastructure. In: Proceedings of wireless internet conference, Boston
61. Marfia G et al (2007) Evaluating vehicle network strategies for downtown Portland: opportunistic infrastructure and importance of realistic mobility models. In: Proceediings of MobiOpp 2007, Porto Rico
62. Mejri N, Kamounh N, Filali F (2010) Cooperative infrastructure discovery through V2X communication. In: Proceedings of 9th IFIP annual mediterranean ad hoc networking workshop, pp 1–8, June 2010
63. Wedel JW, Schunemann B, Radusch I (2009) V2X-based traffic congestion recognition and avoidance. In: Proceedings of 10th international symposium on pervasive systems, Algorithms and networks, pp 637–641, Dec 2009
64. Jeongwook S, Kyungwon P, Wongi J, Dong KK (2009) Performance evaluation of V2X communications in practical small-scale fading models. In: Proceedings of IEEE 20th international symposium on personal, indoor and mobile radio communications, pp 2434–2438, Sept 2009
65. Mostafa A et al (2011) A V2X-based approach for reduction of delay propagation in vehicular ad-hoc networks. In: Proceedings of international workshop on seamless connectivity in vehicular networks, Saint-Petersburg, Russia, Aug 2011

66. Mostafa A, Vegni AM, Oliveira T, Little TDC, Agrawal DP (2012) QoSHVCP: hybrid vehicular communications protocol with QoS priorization for safety applications. ISRN Commun Networking 2012:14
67. Mershad K, Artail H, Gerla M (May 2012) ROAMER: roadside units as message routes in VANETs. Ad Hoc Netw 10(3):479–496
68. Zhao J, Arnold T, Zhang Y, Cao G (2008) Extending drive-thru data access by vehicle-to-vehicle relay. In: Proceedings of ACM international workshop on vehicular inter-NETworking, VANET, San Francisco, CA, USA, Sept 2008
69. Yoo J, Choi SC, Gerla M (2010) An opportunistic relay protocol for vehicular road-side access with fading channels. In: Proceedings of IEEE international conference on network protocols, ICNP, Kyoto, Japan, Oct 2010
70. Campolo C, Molinaro A (2011) Improving V2R connectivity to provide ITS applications in IEEE 802.11p/WAVE VANETs. In: Proceedings of ICT 2011, Ayia Napa, Cyprus
71. Vegni AM, Vegni C, Little TDC (2010) Opportunistic vehicular networks by satellite links for safety applications. In: Proceedings of the fully networked car workshop. Geneva International Motor Show, Geneva, Switzerland, Mar 2010
72. Murthy ASN, Mohle HR (2000) Transportation engineering basics. American Society of Civil Engineers, New York

Chapter 8
Probabilistic Routing Schemes for Ad Hoc Opportunistic Networks

Vangelis Angelakis, Elias Tragos, George Perantinos and Di Yuan

Abstract In this chapter, we review probabilistic schemes for routing in ad hoc opportunistic networks. The literature holds a vast number of publications. For this, we select only a handful of the most influential schemes that have appeared and conclude by giving a brief overview of a recently proposed simulation framework whose goal is to provide understanding of how lower layer conditions affect the decision of the probabilistic parameters in such a scheme.

Keywords Cross-layer · Forwarding · Routing · Sensor network · Vehicular network · Protocols

8.1 Introduction

The proliferation of constantly evolving, consistently cheaper, wireless network interfaces has created the opportunities and demand for wirelessly networking devices and exciting new applications. With mass-market radio technologies such as 3/4G, IEEE 802.11, Bluetooth, as well as other radio solutions such as low power radios designed for use in sensor networks, it has become possible to equip practically any device with wireless networking capabilities. As a result, catch-phrases such as the

---

V. Angelakis (✉) · D. Yuan
Department of Science and Technology, Linköping University, Linköping, Sweden
e-mail: vangelis.angelakis@liu.se

D. Yuan
e-mail: diyua@itn.liu.se

E. Tragos
Institute of Computer Science, Foundation for Research and Technology—Hellas,
Heraklion, Greece

G. Perantinos
Forthnet S.A., Athens, Greece

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_8, © Springer Science+Business Media New York 2013

"Internet of Things" and "machine-to-machine communication" have found their way into the research community as the market demand for global wireless connectivity has gone through the roof. With this ubiquity of wireless-enabled devices, information exchange can start spontaneously, without any prior planning, and for limited time, even where no networking infrastructure is available, thus giving flesh and blood to the ad hoc networking paradigm established in the 1990s [1].

Due to the physical limitations of wireless communication, end-to-end communication between any two nodes in a wireless ad hoc network may have to rely on the relay of messages by intermediate nodes giving rise to multi-hop wireless ad hoc networks. The fundamental requirement in "traditional" networking, for a fully connected path between communication endpoints, naturally carries over to wireless ad hoc networking. Hence forwarding and routing become central mechanisms for end-to-end communication. In multi-hop ad hoc wireless networks, these pose a significant challenge because of the wireless medium. Wireless links tend to be unreliable due to fading and interference [2]. Furthermore, mobility of the networked devices poses an additional challenge. In mobile ad hoc networks (MANETs), a fully connected path between two endpoints may not exist throughout the duration of the communication process. In both such cases, messages may have to be buffered for a long time by intermediate relay nodes, and the mobility of those nodes must be exploited to bring messages closer to their destination by exchanging messages between nodes as they meet. This gives rise to the opportunistic wireless ad hoc networks, a special case of the so-called delay-tolerant networks (DTNs) [3, 4].

In designing a routing scheme for any type of network, but specifically in our case for an opportunistic ad hoc network, there are three key design decisions, namely, *what* to send, *whom* to send it to, and *when* to do so. To understand these issues, we review the answer to these three questions from the classical networking perspective. The answer to the two first questions comes from routing that decides the egress interface to be used to send a packet. The last question is dealt by scheduling mechanisms that decide when to send packets queued for transmission.

Forwarding in classical networking is based on two assumptions: the existence of a connected topology that can be shared among nodes, and the lack of individuality of a node. By the first assumption, a node's local neighborhood knowledge can be spread in the whole network so that *all* nodes know how to reach *any* destination. Routing protocols have the role of spreading this local knowledge in the network and defining the forwarding cooperative behavior, the routing strategy that ensures packet delivery to the end destinations. The second assumption is the lack of individuality of nodes, meaning that they *have to* implement the forwarding behavior determined by the routing protocol and cannot choose their forwarding based on their own interests. Indeed, in a classic network when a router forwards a packet it can assume that it will be forwarded in the next hop. This means that when a node forwards a packet it can let go its interest about it and transfer the responsibility of delivery to the next step down the line. To extend this confidence when a packet has to cross the border of different networks with potentially different policies, the intent of the border node to forward a packet or not is clearly signaled through BGP-like protocols.

It has become rather obvious by now that in opportunistic ad hoc networks, these two assumptions do not in general hold, hence using traditional wired routing schemes that select the best path toward a destination and forward the packet to a specific next hop, may prove ill-suited for such networks utilizing opportunistically the wireless medium and relying on lossy broadcast links, selfish and mobile nodes. In the past decade, a new routing paradigm, namely that of probabilistic routing, has been introduced to cope with unreliable transmissions by taking advantage of the broadcast nature and spatial diversity of the opportunistic wireless medium. In probabilistic routing some of the three key design decisions are made either taking into account the environment and attempting to combat the probabilities of failure in communication, or are randomized in order to enable increased routing flexibility, for example selecting candidate packet forwarders from the group of recipients of the packet even after its broadcast, hence limiting the hard commitment to a predefined fixed path. This probabilistic flexibility enables opportunistic routing to combine multiple weak links to create a reliable route, as well as to exploit unexpectedly long transmissions.

The added forwarding reliability in transmission reduces the retransmission cost, which in turn improves the throughput and energy efficiency. Such routing protocols, which assign forwarding tasks, in a more or less cooperative fashion, require the exchange of node coordination messages, leading to high overhead and increased resource consumption. Furthermore they require global knowledge of the topology which makes them prone to poor performance in the case of misinformation. On the other hand, localized probabilistic forwarding decision protocols, which have been designed mostly for use in sensor application networks, have to trade high performance for robustness and simplicity. In fact some of them partly rely on flooding techniques, thus leading to many potentially unnecessary transmissions.

The aim of this chapter is to revise some of the main contributions and their key findings. We dedicate each of the following sections to a class of probabilistic-based routing protocols for opportunistic routing starting from that of Epidemic Routing in the next section. Despite the fact that Epidemic Routing is not per se a probabilistic routing scheme but it is firstly introduced here for the sake of completeness and continuity. Epidemic routing is a flooding-based, where nodes continuously replicate and transmit messages to those that do not already possess a copy. In its simplest case, epidemic routing essentially reduces to flooding; however, more sophisticated techniques can be used to limit the number of message transfers.

8.2 The Epidemic Routing Scheme

Vahdat and Becker were the first to introduce a bio-inspired [13] routing protocol for intermittently, opportunistically connected mobile ad hoc networks called Epidemic Routing [5]. This protocol relies on the theory of epidemic algorithms [6, 7] by doing pair-wise information of messages between nodes as they get contact with each other to eventually deliver messages to their destination. Hosts buffer received messages even if at the time or reception they have no path to the destination. A so called summary vector is kept by the nodes to index the buffered messages. When two nodes meet they exchange summary vectors. Thus, each node can determine if a newly identified neighbor node has new messages. In such a case, these new messages are requested. This way as long as there is buffer space available nodes (upon request) "infect" each other with the new messages.

Clearly, each message must contain a globally unique identifier to determine if it has been previously seen. Besides the obvious fields of source and destination addresses, messages also contain a hop count field. This similarly to the TTL field in IP packets determines the maximum number of hops a message can be sent (the number of nodes it will infect), and can be used to limit the resource utilization (buffer, and transmission times) of the protocol. Messages with a hop count of one will only be transmitted by the source directly to their final destination, "infecting" no other nodes. The resource usage of this scheme thus is regulated by the hop count set in the messages, and the buffer size set at the nodes. If these are correctly combined to sufficiently large values, the messages may eventually propagate throughout the entire network. Vahdat and Becker have, however, shown that by choosing an appropriate maximum hop count, delivery rates can still be kept high while the resource utilization is lower in the scenarios used in their evaluation.

Evaluation of the performance of epidemic routing schemes initially started through simulation [5, 8]. Markovian models have since been developed to study the performance of epidemic routing [9, 10] as well as the similar schemes [11, 12]. Recognizing the similarities between epidemic routing and the spread of infectious diseases, ordinary differential equation (ODE) models adapted from infectious disease-spread modeling [13] have been used to study the source-to-destination delivery delay under the basic epidemic routing scheme. Markovian models then were used to study other performance metrics. In [14], through ODE modeling, insights were obtained into different epidemic routing schemes. In particular, some basic rules of thumb for configuring these schemes were identified, the existence of a linear relation between total number of copies sent and the buffer occupancy under certain schemes was shown. Finally, analysis of buffer-constrained epidemic routing suggested that sizing node buffers to limit packet loss is not vital provided that appropriate buffer management schemes are used.

8.3 PRoPHET

Epidemic routing is particularly resource demanding because it makes no attempt to eliminate replications that are unlikely to improve the delivery probability of messages, on the contrary, it relies on the reverse principle of generating as many duplicates as possible. This strategy is effective if the encounters between mobile nodes are *purely random*, but in realistic situations, encounters are rarely totally random. According to the notion of the data mule, mobile terminals carried by humans move within a pseudo-random fashion within some social habits and accordingly tend to have greater probabilities of meeting certain other nodes. The Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) protocol, by Lindgreny, Doria and Scheleny [15], introduced algorithms that attempt to exploit the pseudo-randomness of real-world encounters. Although random mobility model is popular in evaluations of mobile ad hoc protocols, real users are indeed not likely to move around randomly, but rather move in a predictable fashion based on repeating behavioral patterns such that if a node has visited a location several times before, it is likely that it will visit that location again.

A probabilistic metric called Delivery Predictability, $P(M; D)$ (with probability values in [0; 1]), is established at every node M for each known destination D. This indicates how likely it is that node M will be able to deliver a message to that destination. The operation of PRoPHET is similar to that of Epidemic Routing. When two nodes meet, they exchange summary vectors which in this case also contain the delivery predictability information stored at the nodes. This information is used to update the internal delivery predictability vectors as described below, and then the information in the summary vector is used to decide which messages to request from the other node based on the forwarding strategy used.

When some message arrives at a node, there might not be a path to the destination available so the node have to buffer the message and upon each encounter with other nodes, the decision must be made on whether or not to transfer a particular message. Furthermore, it may also be reasonable to forward a message to multiple nodes to increase the probability that a message is really delivered to its destination. Unfortunately, these decisions are not trivial to make. In some cases, it might be sensible to select a fixed threshold and only give a message to nodes that have a delivery predictability over that threshold for the destination of the message. On the other hand, when encountering a node with low delivery predictability, it is not certain that a node with a higher metric will be encountered within reasonable time. Thus, there can also be situations where we might want to be less strict in deciding who to give messages to. Furthermore, there is the problem of deciding how many nodes to give a certain message to. Distributing a message to a large number of nodes will of course increase the probability of delivering a message to its destination, but in return, more system resources will be wasted. On the other hand, giving a message to only a few nodes (maybe even just a single node) will use little system resources, but the probability of delivering a message is probably lower, and the incurred delay high. In the evaluations in this paper, we have chosen a rather simple forwarding strategy—when two nodes meet, a message is transferred to the other node if the delivery predictability of the destination of the message is higher at the other node.

Overall during the opportunistic encounters PRoPHET message replication follows the basic rule to replicate if (a) the new node does not have the message and (b) appears to have a better chance to deliver it.

To determine the delivery predictabilities in each node an adaptive algorithm is used. Node M stores delivery predictabilities $P(M, D)$ for each known destination D. If the node has not stored a predictability value for some destination, then that $P(M, D)$ value is taken to be zero. The delivery predictabilities used by each node are recalculated at each opportunistic encounter according to three rules:

1. When the node M encounters another node N, the predictability for N increases as:
   $$P(M, N)_{new} = P(M, N)_{old} + (1 - P(M, N)_{old}) \times L_{enc},$$
   where L_{enc} is an initialization constant.
2. The predictabilities for all destinations D other than N suffer an ageing:
   $$P(M, D)_{new} = P(M, D)_{old} \times \gamma^{K},$$
   where γ is an aging constant and K is a time factor dependent on the previous aging effect.
3. M and N exchange predictabilities stored and the transitive property is used to update the predictability of destinations D for which N has a $P(N, D)$ value on the assumption that M is likely to meet N again:
   $$P(M, D)_{new} = P(M, D)_{old} + (1 - P(M, D)_{old}) \times P(M, E) \times P(E, D) \times \beta,$$
   where β is a constant scaling factor.

An important drawback of PRoPHET routing, is its inability to support priorities and, as a result, to provide any form of Quality of Service. In this context, all data packets are handled equally and no special treatment can be applied to urgent data. Most important, PRoPHET routing consumes considerable amount of both energy and time for message exchange prior to each transmission.

The N4C (Networking for Communications Challenged Communities: Architecture, Test Beds and Innovative Alliances) EU FP7 project [16] deployed real DTN systems in remote areas in Sweden. Within N4C, an implementation of PRoPHET was evaluated and of the most prominent issues that were unveiled was that the original paper DPs were all very high and that they did not describe well to the actual network topology. A reason for this unpredicted behavior was identified in the transitive update equation of PRoPHET. When the frequency of encounters is not evenly spread over the network, and encounters are frequent enough that the aging mechanism does not reduce the DP rapidly enough, such problems can occur.

Keränen et al. from the Helsinki University of Technology [17, 18], compared PRoPHET to other protocols using various synthetic mobility scenarios. PRoPHET was found to be sub-optimal for Random Waypoint (RWP) types of scenarios, partly because RWP mobility does not provide predictable mobility patterns that PRoPHET can leverage. When more realistic mobility was used, PRoPHET outperformed Epidemic Routing. It is noteworthy that in the more realistic scenarios, the delivery rate of PRoPHET was close to a single-hop direct delivery method. This indicates that the mobility is such that most nodes will eventually meet a given destination, thus not ideal for showcasing the strengths a multi-hop protocol. In [19], the authors of PRoPHET propose an improvement of the protocol with only minor changes in specifications drawn from the experience of the deployed implementations of PRoPHET routing protocol mostly within N4C.

8.4 MaxProp

MaxProp was developed at the University of Massachusetts, Amherst by Burgess et al. [20]. MaxProp is flooding-based in nature, in that if a contact is discovered, all messages not held by the contact will attempt to be replicated and transferred. MaxProp tries to determine which messages should be transmitted first and which messages should be dropped first. Thus addressing the lack of QoS in PRoPHET MaxProp maintains an ordered queue based on the destination of each message, ordered by the estimated likelihood of a future transitive path to that destination. The motivation for it is based on vehicular network or pedestrians. Vehicles can provide substantial electrical supplies and transport bulky hardware, which may be inappropriate for use by non-mechanized peers. A key disadvantage of the former is that the nodes move more quickly, reducing the amount of time they are in radio range of one another. Accordingly, the protocol addresses scenarios in which either transfer duration or storage is a limited resource in the network. MaxProb releases the bias toward short-distance destinations, by using hop counts in packets, serving at the same time as a measure of network resource fairness. Through acknowledgments are propagated network-wide, and not just to the source, MaxProp addresses stale data. Finally, MaxProp stores a list of previous intermediaries to prevent data from propagating twice to the same node. While these ideas were simple, the authors' experiments show they significantly raise the delivery rate and lower latency in a wide variety of scenarios as compared to previous approaches.

The initial MaxProp scenario assumes that each peer has an effectively unlimited buffer for messages produced, but a fixed-size buffer for carrying and relaying messages originated by others. Transfer opportunities are assumed to be limited both in duration and bandwidth. Furthermore, nodes are assumed have no a priori knowledge of network connectivity, no control over their movement, no knowledge of geographic location. The authors argue that as in a real network, the opportunistic communication proceeds roughly through three stages. (1) Neighbor Discovery, where Peers must discover one another before a transfer opportunity can begin; and they do not know when *the next* opportunity will begin. (2) Data Transfer. When two peers meet, the amount of data they can transfer is limited. Peers do not know the *duration* of each opportunity. and (3) Storage management. As packets are received from a neighbor, each peer must manage its finite local buffer space by *selecting packets to delete* according to some criterion or algorithm.

Messages that are destined for a receiving peer are passed up to the application layer and removed from the buffer. Each peer carries all messages until a subsequent meeting occurs. A peer will continue to forward a message to any number of other peers until its copy of the message times out, it is notified of delivery by an acknowledgment, or the message is dropped due to a full buffer. Regarding priorities packets that are ranked with highest priority are the first to be transmitted during a transfer opportunity. Packets ranked with lowest priority are the first to be deleted to make room for an incoming packet. When two packets have destinations with the same cost, the tie broken by giving the packet that has traveled fewer hops higher priority.

When two peers discover each other, MaxProp proceeds to exchange packets along the following steps:

1. all messages destined to the neighbor peer are transferred.
2. routing information is passed between peers: This is done through a vector listing estimations of the probability of meeting every other node.
3. acknowledgments of delivered data are transferred, regardless of source and destination. An acknowledgment consists of a cryptographic hash of the content, source, and destination of each message. (Note that this mechanism aims to clear out buffers in the network of old data at a low overhead cost given that the ACK messages are small—compared to data packets). In the original paper evaluation [20], peers would not spend more than 1 % of the historical average connection duration on sending acknowledgments.
4. packets that have not spent many hops in the network are given priority. This is because estimating the delivery likelihood can favor packets that have a high chance of reaching a destination, causing some packets to never get a chance to be transmitted. Therefore, MaxProp attempts to give new packets a "head start" in the network by giving them a higher priority. The effect of this approach is that newer packets are transmitted at several transfer opportunities when they are generated, thus expanding fast toward the destination. To implement this strategy, MaxProp splits the buffer in two logical sub-buffers, according to whether the packets have a hop count less than a threshold. Packets below the threshold are sorted by hop count.

   Since a static threshold assignment would be arbitrary and might not work in all environments. MaxProp takes an adaptive approach to setting the threshold. In environments where the average number of bytes transferred per encounter is much smaller than the buffer size, MaxProp prioritizes low hop count packets. As the size of transferred batches grows, the threshold is progressively reduced to the difference between the two values. When transfer batch size is larger than the buffer size, the threshold is completely removed since it is no longer of any effect.
5. the remaining, untransmitted packets are sent in an order the Estimating Delivery Likelihood, which in turn is based on variation of Dijkstra's algorithm.
6. packets that have already been sent to the node are not sent again. A hop list in each packet stores peers that the packet has already traversed, including peers to which the current node has sent the packet.

8.5 Parametric Probabilistic Routing

Barrett et al. in [21] motivated by sensor network scenarios having variable network topologies and nodes with a poor network status view proposed a fundamentally different approach to routing that combines the best features of limited-flooding and information-sensitive path-finding protocols toward increasing the reliability and power consumption, while providing some delivery guarantees independent of parameter values or information noise levels. For this they introduced the Parametric Probabilistic Sensor Network Routing protocols family, which improves the performance of controlled flooding methods like the ones presented up to now by making the probability of a message retransmission a function of several parameters. In a nutshell, Parametric Probabilistic Sensor Network Routing protocol behavior is completely described by the retransmission probability function: each node in the network—upon receiving a packet—retransmits the packet to its neighbors according to a probability function. While this principle is straight-forward, care must be taken not to overload the probability function: the retransmission probability could depend on parameters as diverse as the numbers of copies of the packet already received by a certain node or the distance to the destination. Keeping routing lightweight requires the retransmission probability to be as simple as possible, both in terms of complexity it takes to compute as well as in estimation of parameters that the function depends on.

Two variations of Parametric Probabilistic Sensor Network Routing are proposed. The first is the *Destination Attractor*, where retransmission probability function depends on the distance of the source node to the destination, counted in hops and the hop-distance from the node currently holding a packet to the destination. The rule of thumb behind it is simple: if the packet is getting closer to the destination, then its retransmission probability should be increased.

The second variation named *Directed Transmission* has the probability depending again on these distances taking also into account the number of hops that the packet has already traveled. The retransmission probability function that is used for the *Directed Transmission* protocol is based on the idea that the nodes that lie on a shortest path from the source to the destination should forward packets with a very high probability and the farther away a node is from the shortest path, the smaller its retransmission probability should be.

With respect to misinformation of nodes clearly results in misestimation of the distance parameters. Hence, both schemes introduce a simple exponential in the retransmission probability which serves as a relaxation factor, turning both schemes toward more flooding. The authors argue that this exponential can be viewed as a QoS-depended tunable parameter by the system designer.

Performance measurements of the schemes against each other, as well as against a number of other proposed sensor network routing protocols have been performed through simulation in sensor deployments of dense and sparse urban and random-deployment settings. The investigation focuses on the trade-off between quality of service measures (in particular the fraction of times that at least one packet reaches the destination and the average lag before a packet reaches the destination), load incurred in the network (measured as the number of times that a sensor emits a packet summed over all time steps and nodes) for different levels of noise. The key findings of the presented simulations is that between the two protocols from the Parametric Probabilistic Sensor Network Routing family, Directed Transmission outperforms the Destination Attractor on all levels of noise achieving better quality of service results while incurring lower load in the network. Both Parametric Probabilistic Sensor Network Routing protocols in turn clearly outperform other multi-path protocols. All multi-path protocols though incur more overhead in the network than single-path protocols, for low levels of information noise. For high misinformation levels, single-path algorithms such as simple shortest path routing can fail leading to excessive lag times and even load, while delivering only a small fraction of the transmitted packets.

Overall, results have suggested that multi-path routing algorithms are more robust to misinformation, and can actually have a higher delivery rate, lower lag, and lower load under the same conditions. Surprisingly, these results are largely independent of the chosen setting (dense vs. sparse network, realistic urban distribution vs. random distribution).

8.6 Probabilistic Routing Protocol for Intermittently Connected Mobile Ad Hoc Network (Propicman)

Probabilistic Routing Protocol for Intermittently Connected Mobile Ad hoc Networks (Propicman) was introduced in [22] within the EU FP6-IST project Haggle [23].

It belongs to the so-called fully context-aware routing protocol class. Fully context-aware protocols not only exploit network context (network topology & environment knowledge, application/mission etc) information to optimize routing, but also provide general mechanisms to handle and use this context information. The advantage of this approach is that they can offer much more generic approaches than partially-aware or context-oblivious routing protocols which are customized, respectively, for a specific type of context information or when there is no knowledge of the possibility for a path toward the destination, nor an appropriate next-hop node. Clearly, the solutions presented thus far fall in the latter two cases.

In Propicman, the information at the node, called *node profile*, plays an important role in (i) predicting the mobility of the nodes, and (ii) describing the social environment of the users and their relationships. Similarly to PRoPHET, it is considered that nodes belong to people with repeating behavioral patterns at different timescales (day, week, and month). Hence, if a node has visited a place several times before, it is likely that it will visit this location again in the future.

Thus, relevant information can be deducted by the node profile. Based on the node profiles of its two-hop neighbor nodes, a sender can select, as forwarder(s), the node(s) with the highest probability of delivering the message toward the destination (delivery probability). The delivery probability can be considered quasi-static information; hence, the mobility of a node does not really affect to the validity of Propicman selection.

The authors show that Propicman exploits the mobility, as well as reduces significantly the number of nodes involved in the forwarding process. Thus, the network overhead is at least equally low in comparison with the other dissemination-oriented routing algorithms, such as Epidemic Routing or PRoPHET.

Propicman also takes into account information privacy from design. Most of the solutions proposed to MANET routing force users to share and exchange their information during the routing process, but it is unlikely that people indeed find this comfortable. This is one of the main barriers to MANET acceptance. In Propicman, nodes share their information in a “hidden” format, but this information can still be used for routing. Furthermore, only the destination can read and understand the message content, while this content is hidden from the intermediate nodes.

Overall, to achieve a good tradeoff of efficient resource usage versus number of delivered packets, Propicman attempts to estimate the probability of source and relay nodes to meet the destination of a packet, and thus infer the delivery probability. The most innovative aspects of the approach are:

- Routing with Zero Knowledge: A sender does not need to have any knowledge of other nodes in the network. It bases its routing decisions only on the information it knows about the destination(s).
- User profile: Unlike other protocols working with profiles, Propicman puts the different weights on the attributes of profile (called “*evidence*”). Thus, the user profile is a very flexible element.
- Security within the network: Propicman aims to protect not only the message content, but also destination information within the scope of secure network applications.

When a node S has a message M to send to destination D, it broadcasts to its neighbors M’s routing header. This header contains the information S has about the destination. Based on this information, the neighbor nodes compute their delivery probability. Propicman assumes (rather loosely) that this is independent for each of the nodes. M’s header is then forwarded to the two-hop neighbors of S. S will send the message Monly on the two-hop route(s) with the highest delivery probability, if this is higher than its own. After sending the message content, S keeps the message for the next eventual encounter. Two-hop routes have been shown to be the most suitable choice to exploit routing information with minimal additional costs and risks [22]. In order to compute the delivery probability at each node, a common set of evidences is established; that is the list of evidences used in the network and the related weight (importance of this evidence in the network). Thus, each node, builds its node profile, which comprises the common set of evidences. The node profile is represented by the set of evidence/value couples.

8.7 A Framework for Probabilistic Routing in Multi-Hop Wireless Networks

In [24] we had argued that, at the time, many of the proposed opportunistic protocols demonstrated a lack of concrete understanding of the way key wireless networking primitives and design decisions affect the performance of a probabilistic opportunistic routing scheme. Thus, it seemed unclear to which extent the reported improved performance of those protocols owed to their probabilistic/opportunistic features and to which extent it was affected by other design features that can also be applied to traditional routing schemes. To this end, using a simulation framework, we examined how the forwarding probability decisions and transmission timings affect performance (in terms of delivery probability and delay) and under which channel error conditions and topology density it is beneficial to use probabilistically opportunistic routing instead of traditional (shortest path) routing schemes. Furthermore, we introduced an opportunistic forwarding scheme, whose probabilistic parameters are adjustable to allow for low resource consumption and high delay performance, while being robust to misinformation.

In order to determine which nodes will forward a packet in a stochastic manner, a forwarding probability function is utilized. This function is the same for all nodes in the network. Its role is to map networking metrics to forwarding probabilities to each node that receives a packet. For the framework developed, any cost metric can be used to this end, (provided, the metric has well-defined minimum and maximum values). In order for a node to be able to calculate its routing metric from the destination of the packet, a neighbor discovery phase should take place between the nodes in network, before data packets can be exchanged. We considered this to have taken place offline and do not assume it to be a part of the protocol, although it is an aspect that would increase the protocol overhead. Note that such processes may be prone to misinformation due to the overly distributed nature of the network.

More specifically, the forwarding decision function is (a) non-increasing in the metric, i.e., lower routing cost values should not yield lower probability, and (b) bound, so that the minimum values it assigns are between zero and one. Given the cost metric assumed, the forwarding probability, per received packet on a relay, is assigned with respect to the distance of the node from the packet destination and its relative position to the sender, similarly to PRoPHET. Hence, when a node broadcasts a packet it has to include in the routing header its distance from the destination, so that the nodes that receive it can calculate their forwarding probability for it. A key in the design is that out of all the neighbors of the sender node, the neighbor that is closest to the destination should be assigned a probability equal to one, to ensure the progress of the packet toward the destination. In what follows, we interchange the terms “routing cost metric” and distance.

Having calculated the probability to forward a certain packet received, a relay candidate node should proceed to decide when to do so. A randomized back-off mechanism is then used, where each node calculates a maximum window of back-off time slots and then randomly selects a number of slots from that. After the back-off timer has expired, the node will transmit the packet with the predetermined probability. The focus of this access control-layer process is to differentiate the back-off times of different nodes, focusing more on the ones with high forwarding probability. These highly-likely forwarders contribute significantly to the packet’s progress towards the destination; therefore it is important to avoid collisions between their transmissions, bringing the packet closer to the destination as fast as possible. Therefore, these back-off values are inversely proportional to a node’s forwarding probability.

To acknowledge successful packets, forwarding takes advantage of the broadcast nature of the wireless medium. After transmitting a packet, a node overhears its neighbors' transmissions for a sort amount of time in order to verify that the packet is retransmitted, thus also avoiding collisions. If a transmission of the last sent packet is captured during this monitoring period, then the sender will remove the packet from its queue and continue to transmit the next in line. Otherwise, the sender will retransmit it, as long as a maximum number of retransmissions has not been reached.

Among the adjustable parameters of the proposed scheme, are the shape of the forwarding function, the acknowledgment delay and a TTL parameter on the packets. They, respectively, tend to the number of candidate forwarders, the probability for collisions or duplicates' generation, and the geographical spread of a packet in the network along with the resource utilization.

Simulation results demonstrate that the optimal fashion to adapt to increasing error is to increase the number of forwarders by increasing the slope of the forwarding probability function. In particular, the number of certain forwarders has the most impact on performance and they need to be increased with respect to error conditions. On the other hand, to reduce resource consumption, in terms of packet transmissions, utilizing a step-wise function is a sound approach, which proved robust to metric miscalculations. Finally, there is a tradeoff between differentiating each forwarder's back-off value to reduce resource consumption and reducing delay. Simulation results show that a variable back-off scheme that gives priority, by means of smaller back-off windows, to best forwarders according to their forwarding probability is preferable to a fixed back-off window for all forwarders.

8.8 Discussion

We have revised the most influential probabilistic routing schemes for opportunistic ad hoc networks. We reviewed general principles from Epidemic Routing to forwarding practices in protocols with actual implementation in real-life trial scenarios such as the PRoPHET. We concluded presenting a framework that encompasses the key features of the recent probabilistic routing schemes introducing a simulation framework similar to that of [18]. To develop and evaluate such schemes, a wide host of tools have been thus far utilized from simulation, to Markovian models and ordinary differential equation systems.

Acknowledgments The authors acknowledge the support of the Swedish ELLIIT research center, the Swedish Research Council (Vetenskapsrådet), the EU-FP7 MC-IRSES Project 318992, and the EU-FP7 Marie Curie project MESH-WISE.

References

1. Misra S, Woungang I, Misra SC (eds) (2009) Guide to wireless ad hoc networks. Computer communication network series. Springer, Berlin
2. Rajaraman R (2002) Topology control and routing in ad hoc networks: a survey. ACM SIGACT News. doi:10.1145/564585.564602
3. Cerf CG et al (2007) Delay tolerant network architecture. RFC 4838
4. Wang Y et al (2007) A survey on analytic studies of delay-tolerant mobile sensor networks. J Wirel Commun Mobile Comput 7:1197–1208
5. Vahdat A, Becker D (2000) Epidemic routing for partially-connected ad hoc networks. Duke technical report CS-2000-06
6. Ganesan D et al (2002) An empirical study of epidemic algorithms in large scale multi-hop wireless networks. Technical report UCLA/CSD-TR-02-0013, UCLA Computer Science Department
7. Demers A et al (1988) Epidemic algorithms for replicated database maintenance. ACMSIGOPS Oper Syst Rev 22:8–32 (ACM Press)
8. Juang P et al (2002) Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with Zebranet. In: ASPLOS-X
9. Small T, Haas ZJ (2003) The shared wireless infostation model—a new ad hoc networking paradigm. In: ACM MobiHoc
10. Groenevelt R, Nain P, Koole G (2005) The message delay in mobile ad hoc networks. Perform Eval 62:210–228
11. Small T, Haas ZJ (2005) Resource and performance tradeoffs in delay-tolerant wireless networks. In: ACM workshop on delay tolerant networking
12. Spyropoulos T, Psounis K, Raghavendra CS (2005) Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In: ACM workshop on delay-tolerant networking
13. Daley DJ, Gani J (1999) Epidemic modelling. Cambridge University Press, Cambridge
14. Zhang X et al (2007) Performance modeling of epidemic routing. Comput Netw 51:2867–2891
15. Lindgren A, Doria A, Scheleny O (2003) Probabilistic routing in intermittently connected networks. In: ACM MobiHoc
16. The Networking for Communications Challenged Communities project. http://www.n4c.eu/
17. Karvo J, Ott J (2008) Time scales and delay-tolerant routing protocols. In: ACM MobiCom CHANTS
18. Keränen A, Ott J, Kärkkäinen T (2009) The ONE simulator for DTN protocol evaluation. In: ICST SIMUTools
19. Grasic S et al (2011) The evolution of a DTN routing protocol—PRoPHETv2. In: ACM MobiCom CHANTS
20. Burgess J et al (2006) MaxProp: routing for vehicle-based disruption-tolerant networks. In: IEEE INFOCOM
21. Barrett CL et al (2005) Parametric probabilistic routing in sensor networks. Mobile Netw Appl 10(4):529–544
22. Nguyen HA, Giordano S, Puiatti A (2007) Probabilistic routing protocol for intermittently connected mobile ad hoc networks (PROPICMAN). In: IEEE WoWMoM
23. The FP6-IST HAGGLE Project. http://www.haggleproject.org/
24. Gazoni N et al (2010) A framework for opportunistic routing in multi-hop wireless networks. In: ACM PE-WASUN

Chapter 9
On Performance Modeling of Ad Hoc Opportunistic Routing Protocols

Arka Prokash Mazumdar and Ashok Singh Sairam

Abstract Since the inception of opportunistic routing in ad hoc networks, a number of OR protocols have been proposed. The performance measure of these protocols is predominantly based on simulation. Although simulation provides a simple and economical mechanism to test complex routing algorithms, they have been reported to produce inconsistent results. Simulation-based evaluation of ad hoc routing protocols should, therefore, be complemented with mathematical modeling and verification. OR protocols try to restrict the number of transmissions, therefore, the most important performance evaluation metric of OR protocols is packet transmission. In this chapter we examine the various mathematical evaluation models available in literature. During our study we found that these models do not consider retransmitted packets. We, therefore, propose an analytical model which account for packet retransmissions. The mathematical models are validated using prominent OR protocols. The performance of OR protocols depend not only on design issues like candidate selection and forwarder prioritization but also on network parameters like node topology, node density, etc. While the user has no control on the network parameters the performance of the protocol can be improved by carefully tuning the design issues. As a proof of concept we propose an OR protocol that minimizes packet retransmissions.

9.1 Introduction

Ad hoc wireless networks are emerging as an important means of communication in view of their flexibility and portability [12]. In a wireless ad hoc network nodes interconnect wirelessly through multi-hop routing paths without any centralized control

---

A. P. Mazumdar (✉) · A. S. Sairam
Department of CSE, IIT Patna, Patna, India
e-mail: arka@iitp.ac.in

A. S. Sairam
e-mail: ashok@iitp.ac.in

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_9,

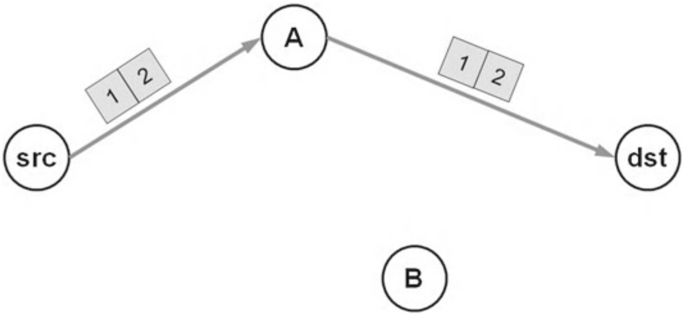


Fig. 9.1 Routing using traditional ad hoc protocols

or fixed infrastructure. All the nodes have similar privileges and they collectively participate in the routing procedures. A great number of routing protocols that have been proposed so far in the literature are based on traditional routing techniques [9, 17]. Although these protocols attempt to optimize different metrics, in general they select a path from source to destination prior to injecting packets to the network and then forwards data through that predetermined route as shown in Fig. 9.1. In this figure the node *src* wants to send a packet to the node *dst*. To do so, node *src* selects the path through *A* to reach *dst*. However, wireless networks by nature have unstable conditions. Other factors which can make a wireless link unreliable are duty cycle of the nodes for preservation of energy and mobility. For example, if the link between *src* and *A* becomes unstable then *src* will be unable to forward data to *dst* as its next hop node in the selected route will be unavailable causing a broken route. In traditional routing to solve this problem the source will reselect the route through *B* and retransmit the packets through this new route. Thus traditional routing protocols have additional overhead for route maintenance.

All wireless transmissions are broadcast. Hence when *src* transmits both *A* and *B* can hear the packets, but since *B* was not chosen as a forwarder it ignored the transmission. If *B* would have buffered the packets and transmitted them when the link between *src* and *A* failed, then the source node could have avoided the overhead of route reselection and packet retransmission. OR protocols have been designed to exploit these broadcast nature of wireless networks. In OR protocols the source node, rather than choosing a single path to the destination, selects a number potential forwarders between the source-destination pair and broadcasts the packet. Among these selected forwarders the node closest to the destination that received the packet successfully forwards it again. This procedure continues until the packet reaches the destination. The paradigm can be explained using a simple example as shown in Fig. 9.2. Here the source node tries to send two packets ‘1’ and ‘2’ to the destination node. Lets assume, as the source node broadcasts the two packets node *A* receives both the packets and node *B* receives only the second packet. Node *B* being closer to the destination than node *A*, it will be first to retransmit and will forward the second packet. As node *A*’s chance comes it will forward the packet ‘1’ as the other

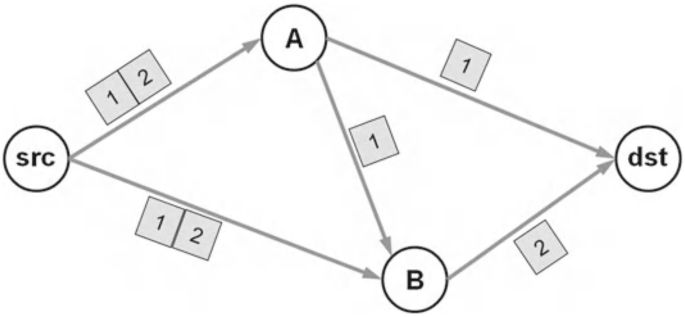


Fig. 9.2 Routing using opportunistic routing protocols

forwarder did not send the packet. This way, as OR does not depend upon a single route, the selected forwarders collaboratively reduce the number of transmissions needed to deliver packets. For these reasons opportunistic routing (OR) protocols gained attention as a procedure to attain better reliability and throughput.

In the remainder of this chapter we first give a overview of the OR routing paradigm for ad hoc wireless networks. We then describe state of the art in performance modeling of ad hoc opportunistic networks. We show that the result of these performance models do not agree with simulation results since they do not consider duplicate transmissions. A new mathematical is introduced to account for such duplicate transmissions. The scope of our model is illustrated by using it to evaluate the classic ad hoc OR protocols ExOR [2, 3]. Based on the findings of our study we finally propose a OR protocol [15] that minimizes transmissions.

9.2 Opportunistic Routing Protocol Paradigm

Opportunistic routing protocols are a relatively new class of routing protocols that tries to exploit the broadcast nature of the wireless medium to improve performance of wireless routing protocols. In this section, we describe the basic principles and then discuss various issues of opportunistic routing paradigm in brief. We first describe the principles of OR protocols with their working procedures and proposed packet formats. Then we discuss about some of their important design and implementation issues in brief.

9.2.1 Principles of OR Protocol

Multi-hop wireless networks have two major differences over the wired networks which has been exploited by OR protocols. First, all pairs of nodes are one-hop neighbors, i.e., all the nodes can sense the transmissions of all other nodes, may be with a very low probability of sensing the packets successfully. Second, all the packets in wireless networks are broadcast, i.e., all the neighboring nodes can sense the packets without depending upon whether they are intended for it or not.

The basic idea behind how an OR protocol works in an ad hoc wireless network [21] is as follows. In a network scenario a source node wants to send packets to a destination node separated by several hops. There will be many nodes between the source and the destination nodes, some among these nodes will try to participate in the routing process. To decide which subset of the nodes will participate in the routing process the nodes will run a decision-making process before starting transmission. The two main steps in an OR protocol are (i) selection of the forwarder list [13] and (ii) prioritization among these forwarders. These steps allow nodes to discover and maintain source routes to destinations in the ad hoc network. All packets destined for the same destination are collected into a batch collectively identified by a unique batch ID. The Forwarder List contains the prioritized forwarder list prepared by the source. All the packets in a batch contains the same set of forwarders list, i.e., the forwarders list for a batch of packets does not change over time and each forwarder keeps a local forwarders list with it copied from the header of one of the packets in the batch. The source and destination are included in the forwarder list as the lowest (forwarder number 0) and highest priority (forwarder number n, where n is the total number of forwarders) node, respectively.

After the source node prepares the forwarder list, it starts transmitting the packets. If the destination node does not receive a packet in the first transmission, the node in the forwarder list which is *closest* to the destination will broadcast the packet if it has received it successfully, otherwise the *next closest node* will try. The idea is to identify a network metric such that the packet will move closest to the destination in each successive transmission. This process of forwarding the packet will continue till the destination node receives the packet successfully.The performance of an OR protocol will depend on how these two steps are performed. The routing issues which we have highlighted earlier needs to be incorporated in these two steps.

The proposed packet format of an OR packet [1, 3] is shown in Fig. 9.3. The Ver field indicated the present version number of the OR protocol which is kept for future use. The fields HdrLen and PacketLen shows the size of the header and the payload, respectively. The IP addresses of the source and the destination nodes are kept in the next two fields namely, Source IP and Destination IP. The batch number to which an OR packet belongs to is indicated by the Batch ID field. Among the next two fields the field BatchSize shows the total number packets in the batch, whereas the field PktNum shows the packet's offset in the batch it belongs to. The next two fields have been included to take care of fragments. The field Forwarder List contains a copy of the list generated by the source node and FwdSetSize shows the size of this forwarders list. The offset of the currently sending node in the forwarders list is shown in the field ForwarderNum. The field Batch Map indicates the best known forwarder that received the packet successfully. The batch map indicates for each packet in the batch, an acknowledgment by the highest priority node that has received a copy of that packet.

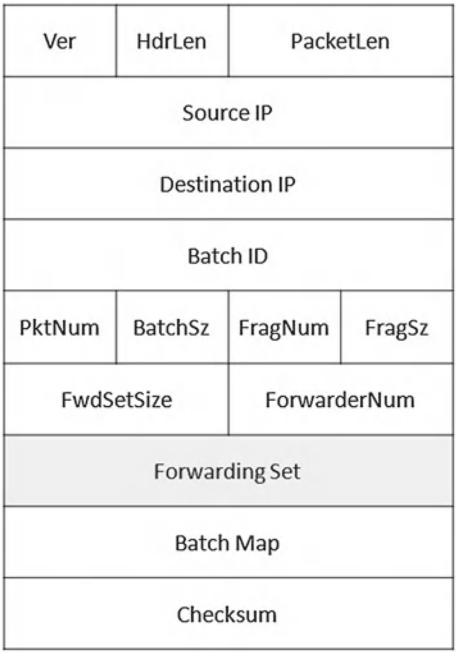
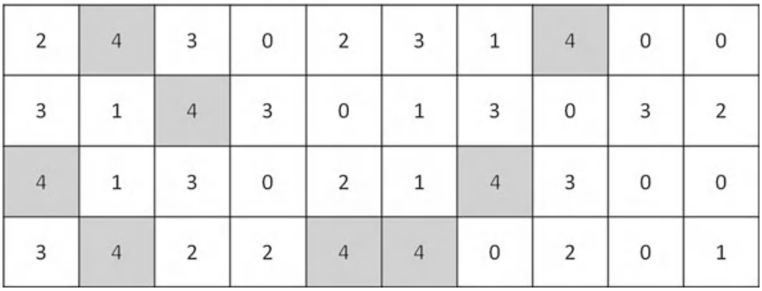


Fig. 9.3 OR packet header format

In Fig. 9.4 we depict a batch map for a batch size of 40 packets. The entries indicate that the first packet is being buffered by the 2nd forwarder, the second and third packet by the 4th forwarder, the fourth packet is still at the source node and so on. The batch map functions as acknowledgement as well as informs about the current status of the packets. A forwarder listens to packets transmitted by its higher priority forwarders, compares the batch map in that packet with its local batch map and updates it if there are any new information. Thus a forwarder check its local batch map and forwards only those that point to its own forwarder number in the list.



2	4	3	0	2	3	1	4	0	0
3	1	4	3	0	1	3	0	3	2
4	1	3	0	2	1	4	3	0	0
3	4	2	2	4	4	0	2	0	1

Fig. 9.4 Batch Map of an OR packet header

9.2.2 Other Design Issues

The performance of an OR protocol not only depends on candidate selection and forwarder prioritization but also depends on a number of other issues. In this section we discuss some of these issues.

9.2.2.1 Scheduling Forwarders

All OR packets are broadcast. In broadcast mode of 802.11 radio specifications while transmitting a broadcast packet the radio does not check for the channel state for other nodes' broadcast. This signifies that the nodes in OR might start broadcasting at the same time causing collisions. Furthermore, in OR a lower priority node is not supposed to send packets already received by the higher priority forwarders which means a lower priority must first get packet status information from the higher priority nodes prior to broadcasting packets. For these reasons OR protocols need to schedule packets in a fashion so that such packet collisions are avoid as well as a higher priority forwarder transmits before any lower priority forwarder.

Most of the OR protocols uses a slotted MAC [3] where the time-slots are assigned according to the priorities of the forwarders and in that slot only that forwarder is permitted to transmit. The slot starting time is calculated as the expected time when all the other higher priority nodes finish their transmissions may lead to either wastage of bandwidth and time slots, or cost duplicate packet retransmissions. To eliminate this need many OR schemes use network coding [4, 11]. However, it has certain disadvantages like high computational complexity, ensuring uniqueness of the coefficients, etc. High computational power may not be available for all kinds of networks such as sensor networks.

9.2.2.2 Measurement Metric

The key ingredient to the performance of an opportunistic routing protocol is the measurement metric for forwarder set selection and prioritization. For OR protocols a single metric may not be sufficient to capture the intricacy of the underlying network protocol. For instance, a routing protocol with an objective to reduce the number of retransmissions that use hop-count as its metric may select routes for which the delivery ratios are too low. In such cases the metric will actually increase the number of retransmissions to deliver the packets reducing the performance of the protocol. There is a challenge to find a metric or a set of metrics to select and prioritize the forwarders for a given network. The metrics proposed for OR protocols are hop-count and PDR [6], geographical location [24], remaining power level (RPL) [10], expected any-path transmissions (EAX) [23] etc.

A related issue is how to minimize the overhead of computing these metrics or periodically compute these metrics and make them available across the network.

9.2.2.3 Bit-Rate Selection

Wireless routing protocols have multiple available bit rates from which transmitters chooses and agrees upon before starting a transmission. As, the traditional bit-rate selection algorithms [8] require sending packets to a single next hop node, the signal strength to that next hop node is measured and the most suitable bit-rate is selected according to it. In opportunistic routing a packet is forwarded to a group of forwarders. Here the conventional bit-rate selection algorithms will fail since it has to decide a mutually possible bit-rate among all the candidates. For this reason presently most of the OR protocols use a fixed bit rate. Employing an optimum bit rate will greatly influence the performance of OR protocols else all the forwarders may not be able to receive the packets due to disparity in speeds.

9.2.2.4 Buffer Size

In OR protocols forwarders need to buffer the packets. A packet is discarded only when the node makes sure that it no longer needs to forward that packet. This means in an OR network all the nodes need a large buffer size. This requirement can be less expensive for wireless mesh and ad hoc networks but the problem will be more severe for sensor nodes or the networks where the participating nodes have limited buffer sizes. In OR protocols the lower priority forwarders, being closer to the source node, have higher chances to receive the packets than the higher priority nodes. It means, when a forwarderś buffer becomes full, most of the forwarders having a lower priority will also be running out of buffer.

9.2.2.5 Quality of Service (QoS)

For delay sensitive applications timely data delivery is of utmost importance and other optimizations like minimizing packet transmissions and preserving energy are secondary. Implementing QoS features in OR protocols needs to deal with challenges like limited resource constrains and dynamic topology of the networks [22].

9.2.2.6 Cross-Layer Protocol

A number of factors influence the performance of OR protocols such as transmission power, frame size, coding technique etc. These issues are not handled by the network layer but by the lower layers. For example, the transmission power level, which decides the signal strengths of the links to the neighbors [19], is controlled by the physical layer. These signal strengths affect the packet delivery ratios to the neighbors and also influences the network layer in selecting the forwarders as the number of neighbors of a node may increase for enhancing the transmission power level [14].

For these reasons some researchers argue that a cross-layered design can prove to be a better approach. One negative side of a cross-layer approach is that these protocols are hardware specific.

9.3 Performance Measurement Model of OR Protocols

Opportunistic routing tries to limit packet retransmissions by taking advantage of the broadcast nature of the wireless channel. The main performance metric of OR is therefore number of packet transmissions. In this section we study the various performance evaluation framework to model *packet transmissions*. We show that the proposed models do not consider the retransmissions. We propose a mathematical model which incorporates the retransmissions also. As a proof of concept we use the proposed framework to evaluate the classical OR protocol ExOR [2, 3].

9.3.1 Statistical Model

9.3.1.1 Markov Models

In [5] the authors present a *discrete time Markov chain* (DTMC) model which can be used to evaluate and compare OR protocols in terms of expected number of retransmission needed to deliver packets to the destination. The model needs packet delivery ratios between all pairs of nodes and the forwarders list as input to compute the number of transmissions needed per packet.

In this Markov model each state represents a forwarder node and the states are sequenced as the positions of the forwarders in the ordered list with destination being the absorbing state. An absorbing state means that it is not possible to go from that state to any other state in the model. This property of the proposed model makes it a discrete phase-type distribution where the last state is strictly an absorbing state. In OR if a packet is received by more than one forwarders then only the highest priority forwarder is supposed to forward the packets. Forwarders having lower priority than the transmitting node do not buffer the packets. The network in Fig. 9.2 can be modelled using DTMC as shown in Fig. 9.5. Here all of the four nodes represent a state placed according to their distances from destination, where the node *dst* is shown as state ‘4’. The last state being the absorbing state has only a self state transition. For all the other nodes there are transitions only to the nodes that are closer to the destination. The transitions probabilities between the states are given in the network. Assuming the delivery probabilities between two nodes m and n as $p(m, n)$ we can compute the transition probabilities for each state. We can see that p_1 depicts a transition to the best candidate, p_2 as the intermediate node and p_3 is the self loop. As p_1 goes to the best candidate, for instance state ‘1’ goes to state ‘3’ the transition probability can be computed for state 1 as

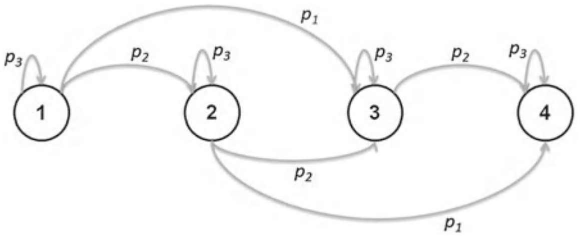


Fig. 9.5 DTMC model of the network given in Fig. 9.2

$$p_1 = p(1, 3) \tag{9.1}$$

The transition p_2 will occur only if p_1 fails to happen which means

$$p_2 = p(1, 2)(1 - p_1) \tag{9.2}$$

The transition p_3 is a special case of p_2 where none of the candidates receive the packet. So, we can compute p_3 as

$$p_3 = 1 - (p_1 + p_1) \tag{9.3}$$

The transition probabilities originating from the other states can also be computed in a similar fashion.

In OR the packets can only move forward toward the destination in each transmission. The transition probabilities in the DTMC model must guarantee this property. That is packets should not move from a state to any of its previous states. In general let us assume a scenario where s is the source node and d is the destination. Let $p(i, j) > 0$ represent the packet delivery probability of the link between the nodes i and j, and $p_{i,j}$ is the state transition probability from node i to j. The ordered set of candidates of vertex i is regarded as $C_{i,d} = \{c_i(1), c_i(2), \ldots, c_i(n_i)\}$ where $c_i(1)$ and $c_i(n_i)$ are the best and the worst forwarders, respectively, to reach the destination node d. The transition probability from a state to any of its previous state should be zero which is stated in Eq. 9.4 [5].

$$p_{i,j} = 0, \quad i < j \tag{9.4}$$

If the highest priority forwarder that can receive packets from node i, namely $c_i(1)$, receives a packet then it will forward the packet with probability 1. This means the transition probability to this node will only depend on the delivery ratio between the two nodes [5] as shown in Eq. 9.5.

$$p_{i,j} = p(i, j), \quad i \neq d, j = c_i(1) \tag{9.5}$$

For all other candidates of current sender i, their state transitions will depend upon the probability of the higher priority forwarders. Specifically, a candidate will forward the packet only if none of the higher priority nodes received the packet successfully which leads to Eq. 9.6 [5]. A special case is where the packets stay in the same state which mean none of the forwarders receive the packet [5] as shown in Eq. 9.7.

$$p_{ij} = p(i, j) \prod_{l=1}^{k-1} (1 - p(i, c_i(l))), \quad i \neq d, j = c_i(k), k = 2, \ldots, n_i \tag{9.6}$$

$$p_{ii} = 1 - \sum_{l \in C_{i,d}} p_{i,l}, \quad i \neq d \tag{9.7}$$

The destination node, being the absorbing state, will only have a self state transition with probability 1 as shown next [5].

$$p_{ii} = 1, \quad i = d \tag{9.8}$$

The model assumes that the forwarder list maintains a strict order, that is, no two candidates can have same priorities. This assumption ensures loop-freeness among the candidates. With the assumptions as stated above the transition matrix for the Markov chain [5] can be given as

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1N} \\ 0 & p_{22} & p_{23} & \cdots & p_{2n} \\ 0 & 0 & p_{33} & \cdots & p_{3N} \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{T} & t \\ 0 & 1 \end{bmatrix} \tag{9.9}$$

Here $\mathbf{T}$ shows the transition probabilities before a packet reaches the destination, and $\mathbf{t} = \begin{bmatrix} p_{1N} & p_{2N} & \cdots & p_{N-1N} \end{bmatrix}^T$ shows the probabilities to directly reach the destination node from $1, \ldots, N-1$.

Using the above equations for the state transition probabilities the model can compute retransmissions for a certain forwarders list using the following recursive Eq. [5]:

$$E[X_i] = \frac{1 + \sum_{l=1}^{n_i} p_{il} E[X_l]}{1 - p_{ii}}, \qquad i \neq d. \tag{9.10}$$

Here X_i is a random variable that represent the number of transitions from state i till total absorption occurs, p_{il} is the probability of transition from state i to state l with p_{ii} being the probability of detaining in the same state i. The above equation is applicable to all the states except the absorbing state (destination node). As it does not transmit any data packet the absorbing state holds the equation $E[X_d] = 0$.

The main problem with the Markov is that it assumes all the nodes have knowledge of the packets received by the higher priority nodes, i.e., it does not consider duplicate retransmissions. Duplicate retransmissions occur in OR due to the failure in receiving updated acknowledgments by the lower priority forwarders. This mainly happens when the forwarder set is not strategically chosen and a lower priority node does not hear from its following higher priority forwarder.

9.3.1.2 Binomial Model

In [15] the authors use binomial distribution to model OR protocol. This model considers the duplicate packets generated due to loss of acknowledgments. Let $\overrightarrow{V} = \{v_0, v_1, v_2, \ldots, v_i, \ldots, v_n\}$ be a ordered forwarders list with v_0 and v_n as the source and destination node respectively. Thus v_0 will have the lowest priority and v_n will be the highest priority forwarder. We assume that, for any three forwarders v_i, v_j, and v_k (arranged according to their increasing order of priorities), if node v_i transmits a number of packets and v_j and v_k listen to the transmission, the number of packets v_k successfully receives is a subset of the packets received by v_j. Let the packet delivery probability (PDP) between two nodes v_i and v_j be p_{ij}. We also assume that all the links between nodes are bidirectional, that means $p_{ij} = p_{ji}$ in our case. Further, we assume that PDPs of the links are constant during the transmission of the whole batch of packets.

Let $N(v_i)$ denote the number of packets received and $\lambda(v_i)$ is the number of packets to be transmitted by node v_i. The number of packets to be transmitted by the node v_i ($\lambda(v_i)$) is the difference between the number of packets it received from lower priority nodes and the number of packets for which acknowledgements are received from higher priority nodes. Let us denote the number of acknowledgments received as $\lambda'(v_i)$. We can define $\lambda(v_i)$ as [15]

$$\lambda(v_i) = N(v_i) - \lambda'(v_i) \tag{9.11}$$

In opportunistic routing first turn to retransmit is given to the highest priority forwarder. After this node finishes transmission the lower priority nodes are given chances one after another on the basis of their priority with the source node being the last node to retransmit. This process is repeated till all packets have reached the destination. We define *round* as the retransmission period starting from the highest priority forwarder to the end of transmission of the source node. To send a batch of packets one or more number of rounds of retransmissions may be needed.

Let X_{ij} be a random variable to denote the expected number of packets received by node v_j from node v_i where $\lambda(v_i)$ is the number of packets transmitted by node v_i with probability $p_{i,j}$. The number packets successfully received (X_{ij}) may be anything from 0 to $\lambda(v_i)$. We use binomial distribution to derive the expected number of packets received by node v_j from node v_i. The binomial distribution is the discrete probability distribution of the number of successes in a sequence of n independent yes/no experiments, each of which yields success with probability p. In our case, each packet transmitted by v_i is an experiment and it is a success with a probability p_{ij} if v_j receives a packet successfully. So the expected number of packets received by node v_j from node v_i can be calculated as [15]

$$E[X_{ij}] = \sum_{k=1}^{\lambda(v_i)} k.b(k; \lambda(v_i), p_{ij}) \tag{9.12}$$

where $b(k; \lambda(v_i), p_{ij})$ is the binomial distribution that gives the probability of receiving exactly k packets out of $\lambda(v_i)$ transmitted. The total number of packets received from all the lower priority nodes [15] can be computed as

$$N_n(v_j) = N_{n-1}(v_j) + \sum_{i=0}^{j-1} \sum_{k=1}^{\lambda_{n-1}(v_i)} E_{n-1}[v_i] \tag{9.13}$$

that is the sum of total number of packets received till the last round and the packets received in the last round.

In OR receiving only one packet from a forwarder is enough to receive all the acknowledgments from that node. The probability of receiving at least one packet by v_i out of $\lambda_n(v_j)$ packets transmitted by v_j can be computed as $\left\{1 - (1 - p_{ij})^{\lambda_n(v_j)}\right\}$. Using this expression the number of acknowledgments can be formulated [15] as

$$\lambda'_n(v_i) = \sum_{j=i+1}^{n} \left\{1 - (1 - p_{ij})^{\lambda_n(v_j)}\right\} N_n(v_j) \tag{9.14}$$

where $N_n(v_j)$ is the number of packets transmitted by v_j in nth round.

So, for all the nodes in the forwarder list we can define the total number of retransmissions for a round [15] as

$$R_n = \sum_{i=1}^{s} \left\{N(v_i) - \lambda'(v_i)\right\} \tag{9.15}$$

The above equation shows the sum of the transmissions performed by all the participation nodes in a certain round. This equation (6) can be extended to calculate total number of retransmissions needed to deliver a batch of packets [15] as follows.

$$R_{total} = \sum_{n=1}^{\infty} \sum_{i=1}^{s} \left\{N_n(v_i) - \lambda'_n(v_i)\right\} \tag{9.16}$$

Equation 9.16 can be used to compute the number of retransmissions for any OR protocol. The value for n is taken from 0 to ∞ as the number of rounds to be taken to deliver the packets is not known a prior. This will not affect the model by getting into an infinite loop as the simulation will stop when, after some rounds, the number of packets transmitted becomes zero and the value of R_{total} becomes constant.

To analyze a protocol in terms of expected number of retransmissions per forwarder [15] we can use the following equation:

$$\lambda_{total}(v_i) = \sum_{n=0}^{\infty} \left\{ N_n(v_i) - \lambda'_n(v_i) \right\} \tag{9.17}$$

Here, Eq. 9.17 computes the total transmissions by node v_i until all the packets are received by destination. The range of n here is also from 0 to ∞ and will terminate after some rounds for the same reason as for Eq. 9.16.

9.3.2 Analytical Model

In [23] (EAX) the authors provided a model to compute the expected number of retransmissions for OR protocols given the forwarders list and the packet delivery probabilities between the forwarders. The authors, however, did not use the model for comparing OR protocols. Rather, they used it as a metric to choose the forwarder list.

The model assumes that the probability of delivery of an acknowledgment is same as delivering a packet as the acks (robust acknowledgement or *RACK*) are piggybacked in the data packets. The model also assumes that the packet size does not increase due to inclusion of RACK and a certain number of repeats of an ack is referred to as RACK size. Hence, if packet delivery probability of a packet is a then delivery probability of a RACK of size n is $1 - (1 - a)^n$. In OR a forwarder listens to the higher priority nodes to receive the RACKs to get the latest informations of the packets' positions. The model assumes that when a node forwards a packets all the higher priority forwarders that received the packet successfully responds with RACK. If this sender does not receive an ack from any of the higher priority nodes it retransmits the packet again. This means if the node fails to receive the RACK despite some higher priority forwarders received the packet, the node will retransmit the packets wrongly.

Let s and d be the source and destination nodes and $C^{s,d}$ is the set of candidate next hops from s to d where c_i is the candidate with priority i. Let us denote f_i as the packet delivery probability from source s to the candidate c_i and a^i_j as the delivery probability of RACK from candidate c_j to c_i. Hence the probability that c_i misses the RACK from c_j will be $f_j(1 - a^i_j)$. The model also assumes that the node c_i can also get informed of the RACK through another intermediate higher priority node c_k having a priority less than that of c_j. The probability of not receiving RACK through c_k can be given as $(1 - a^k_j a^i_k f_i)$. Therefore, the probability that c_i does not receive the RACK from c_j directly or through any of the intermediate candidate nodes is $f_j(1 - a^i_j) \prod_{k>i}^{k<j} (1 - a^k_j a^i_k f_i)$ where $k > i$ means c_k has a priority higher than that of the candidate c_i. Using these probabilities the probability λ_i that candidate c_i does

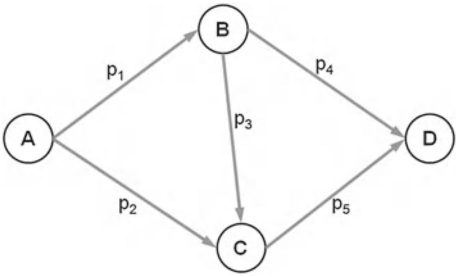


Fig. 9.6 A sample OR scenario where A and D are source and destination, respectively, with B and C as intermediate nodes

not receive RACK from any of the higher priority nodes [23] can be expressed as follows.

$$\lambda_i = \prod_{j>i} (1 - f_j + f_j(1 - a_j^i) \prod_{k>i}^{k<j} (1 - a_j^k a_k^i f_i)) \tag{9.18}$$

The above equation gives a close approximation when the number of candidates in the set is three or less as it only considers one intermediate candidate node through which a lower priority candidate can receive RACK from a higher priority candidate node. However, the formula is not able to accommodate the cases where the RACK reaches to candidates through more than one intermediate nodes. In such cases the model could generate a higher value than it should be, however, as inter-node delivery probability becomes higher the chances of receiving RACK from atleast one candidate also turns high. In cases where the RACK delivery probability is 100 %, the number of duplicate packets will become zero. For those instances the λ_i can be written as $\lambda_i = \prod_{j>i}(1 - f_j)$.

Let us consider the scenario depicted in Fig. 9.6 where node A wants to send some packets to D through intermediate nodes C and D. All the inter-node packet delivery probabilities are shown in the figure. The prioritized order of the forwarders is A,B,C, and D with A having the lowest and D having the highest priority. In the scenario the nodes B and C are the next hop candidates of A. So, the probability of receiving a packet by B (f_B) and C (f_B) from source A can be expressed as

$$f_B = p_1 \tag{9.19}$$

$$f_C = p_2 \tag{9.20}$$

Nodes C and D, being the possible next hop forwarders of B, are able to send RACK to B. But, as D is not a possible next hop for A, D cannot receive the packet. Hence the RACK delivery probability of B from C (a_C^B) with a RACK size n [23] can be given as

$$a_C^B = 1 - (1 - p_3)^n \tag{9.21}$$

which lets us derive the probability that the node B does not get RACK after the packet reception by C as $f_C(a_C^B)$. This way, all the probabilities can be computed and finally the expected number of retransmissions can be obtained for the network using the model.

Finally, the expected number of retransmissions needed to deliver a packet from source to the destination [23] can be gives as

$$EAX(s,d) = S(s,d) + Z(s,d) \tag{9.22}$$

$$S(s,d) = \frac{1}{1 - \prod_i (1 - f_i a_i)} \tag{9.23}$$

$$Z(s,d) = \frac{\sum_i \lambda_i f_i EAX(c_i,d)}{1 - \prod_i (1 - f_i)} \tag{9.24}$$

where a_i is the delivery probability of RACK from c_i to s. The expected number of retransmissions needed to send a packet from s to atleast one of the forwarders and successfully receiving back the acknowledgment is shown in Eq. 9.23, whereas Eq. 9.24 is a recursive formula that shows the number of transmissions needed to deliver the packet to the destination in turn.

If it is assumed that the RACKs are always successfully received by all the nodes then the previous equation can be simplified as follows [23].

$$EAX(s,d) = \frac{1 + \sum_i EAX(c_i,d) f_i \prod_{j>i} (1 - f_j)}{1 - \prod_i (1 - f_i)} \tag{9.25}$$

EAX considers only the one-hop or two-hop RACK reception by a forwarder where the RACK can arrive from a node through multiple number of intermediate nodes. That means, the model can properly model the scenarios where the number of forwarders in the list is three or less. It will present an approximation where the number of forwarders is four or more. In such cases the model will give higher value than expected.

9.4 Evalaution of Ad Hoc Opportunistic Routing Protocols

In the past few years several OR protocols are proposed by the researchers for different type of networks with different network objectives. Extremely opportunistic routing (ExOR) protocol, being the first implementable OR protocol, we discuss it's main features and procedures. Next, we use this protocol to validate and evaluate the binomial model and the DTMC model with simulation results.

Algorithm 1 Forwarder selection algorithm for $ExOR(s, d)$

```
Input: $s, d, N(cand)$
Output: $Fwd(s, d)$
$Fwd(s, d) \leftarrow \phi$
$weight(s) \leftarrow ETX(s, d)$
$visit \leftarrow s$
for all $n$ in $visit$ do
  if $n == d$ then
    $Break$
  else
    for all $cand$ in $N(n)$ do
      if $cand == d$ then
        $weight(cand) \leftarrow 1$
      else
        $weight(cand) \leftarrow ETX(cand, d)$
      end if
    end for
    $N(n) \leftarrow sort(N(n))$
    for all $cand$ in $N(n)$ do
      if $weight(n) > weight(cand)$ then
        if $ret(cand) > 10$ then
          $Fwd(s, d) \leftarrow Fwd(s, d) \cup cand$
          $visit \leftarrow visit \cup cand$
        end if
      end if
    end for
    $visit \leftarrow visit/n$
  end if
end for
```

9.4.1 EXOR

ExOR [2, 3] is the seminal opportunistic routing protocol designed for multi-hop wireless mesh networks. This is the first implementable OR protocol designed by Biswas et. al. [3]. In ExOR the source node builds the forwarders list prior to sending the packets in the network and measures the logical distances of each of the candidate nodes to the destination to select and prioritize the forwarders. After selecting the forwarders the list is included in the header of each of the packets to be transmitted and then the packets are transmitted in batches. All the nodes in the forwarders list receive the packets, buffer them, and wait until its turn to transmit packets. When a node's turn to transmit comes it checks whichever packets are already received by a node with a higher priority than it's own and suppresses those packet from transmitting. This procedure repeats until 90 % of the packets reach the destination node.

Algorithm 1 shows the procedure for forwarder selection for ExOR. The basic idea of this procedure is to run a simulation will all possible candidate nodes and select those having a chance of forwarding atleast 10 % of the total number of packets (batch size). The algorithm first finds the weights for all the candidates starting from source s upto the destination d using the ETX metric where a lower weight means a shorter distance to the destination. Among the traversed candidates, those who has a higher weight to the destination than their previous hop node (which means the previous hop node is closer to the destination than this node), are prohibited to become a forwarder first. The rest of the candidates are eligible to become a forwarder, however, selecting all of them may reduce the performance of ExOR by introducing delays due to the scheduling strategy. So, to eliminate this problem the algorithm chooses only the nodes that are expected to send atleast 10 % of the total packets. The function $rets(x)$ finds the number of packets expected to be forwarded by x if the batch size is 100. The candidates for which this function returns a value larger than 10 are finally selected as forwarders.

9.4.1.1 Comparison and Validation Of Mathematical Models

In this section we compare the DTMC model with our proposed binomial model. We implemented EXOR [3] in OMNet++ [16, 20]. We take scenarios of 20 nodes and simulate ExOR with different source destination pairs. The nodes in the scenarios are scattered over a 1500 by 1500 m^2 open area. We consider log normal shadowing model as our path loss model. The nodes in the area are assumed to be stationary over time and there are no other traffics than the data packets streamed by OR protocol. We make these assumptions as we want to compare the simulation results with the results obtained by the mathematical model where the models do not consider those simulation factors.

The experiment was repeated for different batch sizes and in each run of the experiment we computed the total number of packet transmitted by all the forwarders. Further, during these experiments we also acquired the packet delivery ratios between each pair of nodes and the forwarders lists which are required as input for the mathematical models. These parameters were fed into the mathematical models and the expected number of transmissions were computed. While measuring the results we assume that there are no background traffics in all the scenarios and there is no internode interferences in case of mathematical models.

The results depicted in Fig. 9.7b shows that all the three methods produce similar results and the mathematical models give close approximations of the simulation result. Whereas, in Fig. 9.7a, it can be seen that our proposed model gives better approximated results than the DTMC model. It is worth nothing the in all the three models the results produced by DTMC model is little less than the other two results for all three figures where in Fig. 9.7a the differences are more. This difference is caused due to the assumptions made by the models. The DTMC model assumes that all the nodes have the knowledge of the receptions of packets by the higher priority forwarders, thereby not retransmitting those packets falsely (duplicate packets). In DTMC the model computes the average number of transmissions required to dispatch one packet to the destination. The number of transmissions required for a batch of packets is obtained by simply multiplying this average value with the batch size. Hence, the results of this model is linear as can be seen from the figures.

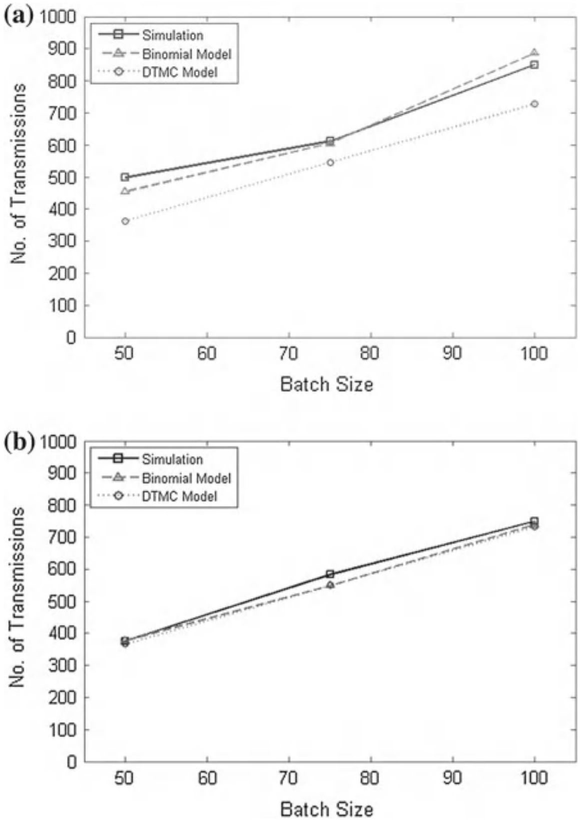


Fig. 9.7 Expected number of transmissions for different number of batch sizes and different forwarders lists for ExOR

Duplicate retransmissions occur in OR due to the failure in receiving updated acknowledgments by the lower priority forwarders. This mainly happens when the forwarder set is not strategically chosen and a lower priority node does not hear from the next higher priority forwarders, and so does not acquire the best positions of the packets. For this reason the nodes may wrongly retransmit packets already held by the higher priority forwarders causing duplicate retransmissions. When two adjacent priority forwarders are not in transmission range of each other, the lower priority node is not able to listen to the next higher priority node, or unable to fetch the acknowledgments from the higher priority forwarder which may inherit the problem stated above. The DTMC model, as it assumes all the acknowledgments are always propagated successfully to the lower priority forwarders it is unable to incorporate the number of duplicate packets in its computation, Most of the forwarder selection algorithms proposed by many authors do not try to actively reduce the number of duplicate packets generated due to miscommunication between forwarders. Furthermore, as the number of duplicate packets mainly increase due to the divergent paths chosen in the forwarders list, the results of the DTMC model is likely to deviate more and give smaller results with the selection of more diverge routes in the list.

In our proposed model as it considers the probability of acknowledgment propagation among the lower priority forwarders too apart from the considering only the internode delivery ratios, the results produced by the model includes the expected number of duplicate packets produced in the simulation life-time. Hence our mathematical result complements the simulation output.

9.5 Minimizing Duplicate Packet Transmissions

In the previous section we had seen that selecting forwarders on divergent paths lead to an increase in the number of duplicate packets retransmitted. In this section, we propose an OR protocol where we consciously try to select forwarders that lie on the same path to the destination. We show that such a protocol minimizes the duplicate packets and therefore reduces transmission.

9.5.1 PBFS

In PBFS [15] the authors have tried to reduce the number of duplicate packet transmissions. The basic idea of PBFS is to increase the chance of successful communication between adjacent priority forwarders so that acknowledgments propagate more reliably.

Let $\overrightarrow{V} = \{v_0, v_1, \ldots, v_n\}$ be an ordered set, sorted according to the increasing orders of priority with source node v_0 having the least priority and destination having the highest priority. We define ℓ as a function that assigns weights to the nodes and links, where $\ell[v]$ means the weight of node v and $\ell(u, v)$ means the weight of the link (u, v). The weight metric can be logical distances between nodes, hop counts to the destination, etc. The operator $\preceq$ is used to compare the priorities among two nodes in term of their weights given by the weight function. A node u is said to be preferred over v if $\ell[u] \preceq \ell[v]$. The set $\overrightarrow{V}$ thus denotes the forwarders list of source node v_0 for destination v_n. We also define $N(v_i)$ as the set of higher priority nodes which are in the transmission range of v_i. In this work our aim is to compute the set $\overrightarrow{V}$ such that the number of retransmission is reduced.

The basic steps of any OR protocol are (1) Candidate list selection, (2) Forwarder set selection, and (3) Prioritization of the forwarders. All of these three steps are discussed next.

9.5.1.1 Candidate List Selection

The protocol first finds all possible paths to the destinations and builds a tree using them. This tree, termed as a *full path tree*, consists of all possible paths from source to the destination node where source is considered as the root and all the leaves are the destination node. The full path tree can be trivially built by starting with the source node (v_s) as root and selecting as its child the nodes in the set $N(v_s)$. The process is repeated recursively for each child node until the destination node is reached. However, the full path tree does not ensure loop-free paths and there may exists some paths that direct the packets away from the direction of the destination node reducing the efficiency of the protocol. Just like conventional routing algorithms [7] clear bad route from its topology table, the protocol also discards the subpaths that create loops or may reduce efficiency.

To prune the full path tree as stated above PBFS selects a node as a potential forwarder (candidate) only if it has a shorter distance than its parent node in the tree. This cropped tree is termed as a candidate tree as each of the nodes in this tree is candidate to become a forwarder. Since our aim in this work is to reduce the number of retransmissions, the protocol chooses inverse of PDP that increase with decrement of the PDP as its weight function ℓ (metric) to assign weights to the nodes. That is,

$$\ell' \propto \frac{1}{PacketDeliveryProbability} \tag{9.26}$$

The weight function ℓ used here can be defined as $\ell\,[u] \preceq \ell\,[v] \Rightarrow \ell\,[u] \leq \ell\,[v]$, which means a lower weight denotes a higher priority. The algorithm to obtain the candidate tree is shown in Algorithm 2.

The procedure starts from the leaf nodes and ends at the root node. Let T denote the full path tree and $T.depth$ be the depth of the full path tree T. Here, T at $height$ returns the set of all the node in $level = height$ and the source or root node (v_0) is at level 0. The weight of the leaf nodes will have the lowest weights as they are the destination nodes themselves. Let us also denote $M(v)$ as the candidate tree or a candidate subtree rooted at a node v. The symbol $\oplus$ is defined as an operator that merges two subtrees to create a new one. The algorithm iteratively adds the child nodes to the new tree which are logically closer to the destination than their parents. The line $\ell\,[u] \preceq \ell\,[v]$ ensures that this child node has a logical distance to the destination than its parent.

9.5.1.2 Selection of Forwarder List

Once the candidate tree containing the candidate nodes is prepared, the protocol selects the potential forwarders to build the forwarder list to participate in packet forwarding. For this purpose the candidate tree is pruned further to get a more simplified tree containing only the forwarder nodes. Previously as we discussed that if adjacent priority forwarders cannot listen to one another's transmission then the situation can

Algorithm 2 Algorithm to Prepare Candidate tree

```
Input: full path tree $T$
Output: $M(v_0)$
height ← T.depth − 1; R ← φ
//Initialize the leaf nodes
for all node in T do
  if node is leaf then
    $\ell[node] \leftarrow 1$
  else
    $\ell[node] \leftarrow \infty$
  end if
end for
//Starting from bottom recursively assign weight to each node
while $height! = 0$ do
  for all node $v$ in ($T$ at $height$) do
    //Compute the weight of each node
    $\ell[v] \leftarrow min_{N(v)}\{\ell[u] + \ell(v,u)\}$
    //Nodes with weight less than its parent node are selected as candidate node
    for all node $u$ in $(N(v))$ do
      if $\ell[u] \preceq \ell[v]$ then
        $M(v) \leftarrow M(v) \oplus M(u)$
      end if
    end for
  end for
  $height = height - 1$
end while
$M(v_0) \leftarrow M(v)$
```

increase the number of retransmissions, the main aim of selection procedure is to select the forwarders such that there is at least one path which includes all of the selected forwarders.

In PBFS, the nodes that lie in the shortest path to the destination are at first selected as forwarders. These selected nodes are called as primary forwarders. The set of primary forwarders is the minimum number of forwarders required to route packets to the destination. However, the main goal of this protocol is to select some additional nodes as brook (backup) forwarders. The brook forwarders can overhear the communication between the primary forwarders and any packet missed by a primary forwarder will be immediately retransmitted by the brook forwarders, thereby reducing the re-transmission time. Speaking in terms of the candidate tree, it means at each hop there must be exactly one primary forwarder and one or more brook forwarders. The number of brook forwarders to be selected per hop depends upon the improvement they offer in term of weight for the current primary and its parent node duo. The rule used by PBFS to select the brook forwarders is: a candidate node between two primary forwarders has higher chances to be selected as a brook forwarder if the product of weights of links from the candidate to both the primary forwarders is lower. To follow the rule the algorithm performs the following task. Let v_p be a parent node that selects v_i as it's next primary forwarder. We want to select

Algorithm 3 Procedure for Forward List Selection

```
Input: $M(v_0)$ //Candidate-tree
Output: $F(v_0)$
$v \leftarrow v_0$
fwdList $\leftarrow \{v\}$
while $v! = d$ do
    // Select a primary forwarder at each hop
    $TempList \leftarrow \phi$ ; $primary \leftarrow \infty$
    for all node $u$ in $(N(v))$ do
        if $\ell[u] \preceq \ell[primary]$ then
            $M(primary)] \leftarrow M(u)$
        end if
    end for
    $F(v) \leftarrow M(v) \oplus M(primary)$
    fwdList $\leftarrow \{fwdList \cup primary\}$
    // Select brook forwarders at same level
    $\ell[M(primary)] = \phi$
    for all $node$ in $(N(v))$ do
        if $\ell\,[F(v) \oplus M(node)] \preceq \ell\,[F(v)]$ then
            $TempList \leftarrow \{TempList \cup node\}$
        end if
    end for
    //re-order TempList
    sort($TempList$)
    for all $brook$ in $TempList$ do
        if $(\ell\,[F(v)] - \ell\,[F(v) \oplus M(brook)]) > Thresh$ then
            $F(v) \leftarrow F(v) \oplus M(brook)$
            fwdList $\leftarrow$ {fwdList $\cup$ $brook$}
        end if
    end for
    $v \leftarrow primary$
    $height \leftarrow height - 1$
end while
$F(v_0) \leftarrow M(v_0)$
```

a brook forwarder (v_j) (v_j must also be a child of v_p) such that delivery of packets to node v_i is ensured, i.e., we want to maximize the PDP to v_i through v_j. Let the delivery probability of the links $v_p \rightarrow v_i$, $v_p \rightarrow v_j$ and $v_j \rightarrow v_i$ be a_p^i, a_p^j and a_j^i, respectively. For a certain v_j the probability of successfully delivering a packet to node v_i [15] will become

$$P_{v_j}\left(v_p, v_i\right) = 1 - (1 - a_p^i)(1 - a_p^j a_j^i) \tag{9.27}$$

where $P_{v_j}(v_p, v_i)$ is the delivery probability from v_p to v_i through the candidate node v_j.

Algorithm 3 shows the procedure for forwarder list selection. Let us denote the final tree after selection as $F(v)$ where node v is the root node. The procedure starts with the root node and iteratively selects the primary and the brook forwarders from each level. It first selects the root node as the first primary forwarder. The procedure then, from each level it selects, as the primary forwarder of that level, the child node of the previous level primary forwarder having the minimum weight. Next, to select the brooks, the method adds a sibling of this presently selected primary forwarder as brook if it offers improvement in delivery ratio as shown in Eq. 9.27 by atleast a $Thresh$ amount.

9.5.1.3 Comparison and Validation Of Mathematical Models

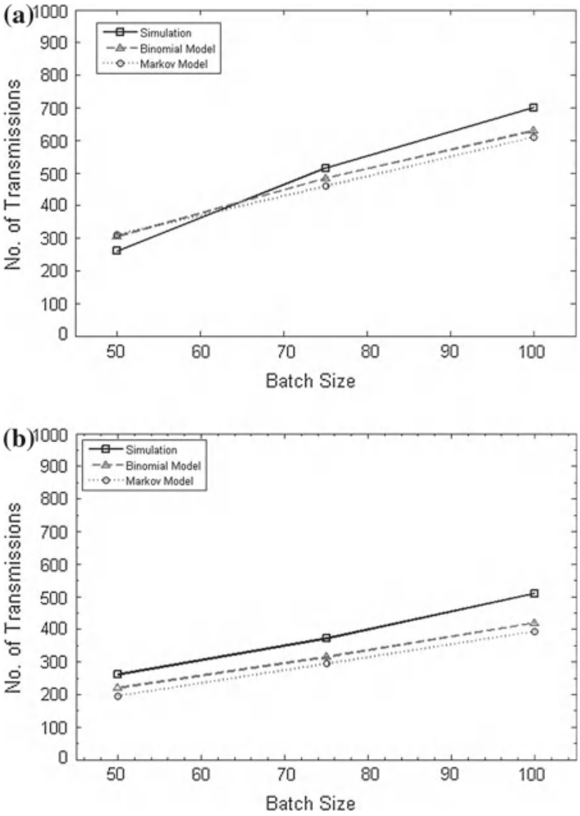
In this section we again compare the DTMC model with our proposed binomial model. We implemented PBFS in Inetmanet framework of OMNeT++. In the same scenario of 20 nodes scattered over an open arena of 1500 by 1500 m^2 as assumed in the previous results we simulate our protocol PBFS and we use these simulation results to validate the mathematical models. Here too we consider log normal shadowing path loss model and assume that the nodes are stationary over time.

We repeat the experiments for different batch sizes and forwarder sets for different source-destination node pair. For each run the total number of transmissions are observed which are to be used for validation of mathematical model. From the simulation environments the packet delivery ratios between all pairs of nodes and the selected forwarders list are recorded which act as the inputs for computations of the mathematical models. As in the simulations for the previous results we assume that there are no background traffics in all the scenarios and there is no internode interferences in case of mathematical models. These assumptions, although not very practical, are essential for the simulations of the mathematical models as they are not able to incorporate those factors.

Figure 9.8 shows the results of our experiments in two different scenarios and forwarders lists with different batch sizes. In Fig. 9.8a we observe that the simulation result shows increment in number of transmissions as the batch size increases with a rate higher than the rates as noted in the results for mathematical models, although the pattern of increment is quite similar. Figure 9.8b shows results of a different scenario where it can be examined that all the three results are very much likely. However, the simulation results show a higher number of transmissions that are needed to deliver packets than that predicted by the mathematical models.

The results of the simulation may drift from the mathematical models for various reasons, specially the factors of the environments that cannot be captured by the mathematical models. As the possibility of interference among the nodes increase with the increment of the number of broadcast packets in the network [18] the simulation results for bigger batch sizes may differ more from the mathematical results. Moreover, these mathematical model assumes that the wireless channel is bidirectional and the delivery ratios in both the directions are equal. This assumption may not be true for simulations and cause differences in the results.

Fig. 9.8 Expected number of transmissions for different number of batch sizes and different forwarders lists for PBFS



9.5.2 Comparison Of ExOR and PBFS

As we validate the mathematical models with the simulation results, it can be seen that the binomial model produces better approximations of the scenarios than the Markov model which claims to produce good results. In this section, we use the binomial model to study and compare both the protocols ExOR and PBFS. We first take a network topology where 50 stationary nodes are scattered over a 1000 by $1000\,\text{m}^2$ square arena. We assume that all the nodes in the network have knowledge of the packet delivery ratios between each pair of nodes. We also assume that there are no background traffic in the networks and no interferences among the nodes. Next, we apply the algorithms as described in Sect. 9.4 to derive the candidate sets of different source-destination pairs for both the protocols. These generated candidate lists and the delivery ratios are then used as input to the binomial model to finally compute the results in term of expected number of transmissions.

The results are shown in Fig. 9.9 and 9.10. Where Fig. 9.9 compares the two protocols in terms of expected number of retransmissions for different source-destination

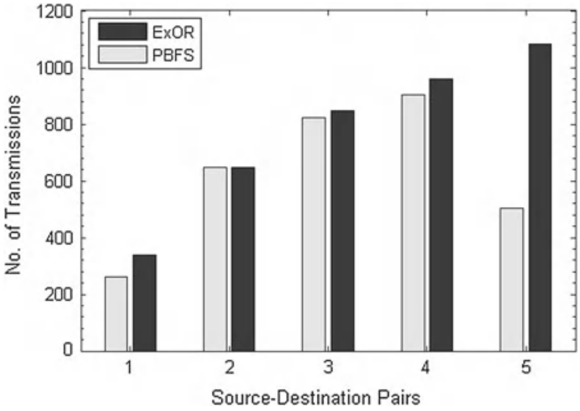


Fig. 9.9 Expected number of transmissions for different forwarders lists

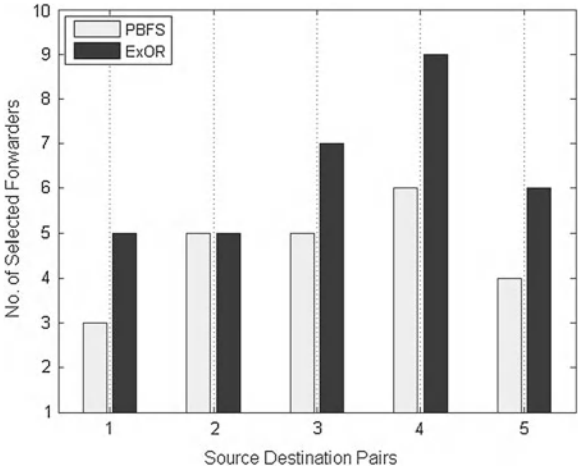


Fig. 9.10 Number of selected forwarders for different forwarders lists

pairs, Fig. 9.10 compares the number of forwarders selected by ExOR and PBFS for the same source-destination pairs. From the Fig. 9.9 we can see that sets 1 and 4 have the similar results that indicate about 10–20 % improvement in PBFS over ExOR whereas, in set 5 we can see PBFS gives around 50 % improvement as compared to the other protocol. Sets 2 and 3 shows results of the forwarder sets where there is not much differences between the two. The set 2 is a special case where both the protocols select the same forwarders list and as a result they produce the same number of retransmissions as can be seen in Fig. 9.9. From Fig. 9.10 we can see that the number of forwarders selected in ExOR is also generally higher than the number of selected forwarders by PBFS. In set 2, as stated before, has the same set of forwarders for both PBFS and ExOR.

The main ground of performance improvement in PBFS is mainly due to the forwarder set selection. As ExOR simply selects the set of forwarders by just running a simulation with the full set of candidates and select those that participate in packet forwarding with a acceptable number of packet forwards, there are chances that the protocol select nodes from different path that are mutually divergent. Hence, the selected forwarder set may give rise to the numbers of duplicated packets generated in the network as discussed in Sect. 9.4.1.1. On the other side, PBFS ensures that the a forwarder is in the communication range of the next higher and lower priority forwarders. This choice of selection improves the chances of successful propagation of the acknowledgments to the lower priority forwarders reducing the expected number of duplicate packet transmissions.

9.6 Problems and Challenges in Modeling OR Protocols

Although packet transmission is a key performance metric of OR protocols, there are several other performance metrics that need to modeled and face several challenges. The performance of an opportunistic routing protocol will depend on the connectivity between forwarders. Modeling contacts will depend on the mobility pattern of the nodes. Connectivity models will also depend on the underlying radio properties of the concerned nodes. Identifying a candidate set that can be mapped to a number of network scenarios is a key research concern. Conserving energy is one of the key goals for mobile nodes. Thus another interesting challenge is to model the energy consumption of an OR protocol. Energy consumption in routing protocols occur mainly during route discovery and route maintenance. In reactive protocols route discovery will be a major factor whereas for proactive protocols it will be route maintenance. Energy consumption during data transfer will be more or less similar in both the cases since data is forwarded through a single path whereas in OR protocols data transfer will consume significant energy since the data is forwarded along multiple paths. The challenge will be to minimize energy consumption by limiting the forwarders but this will conflict with the main goal of OR protocols which is to minimize transmission by choosing a large number of forwarders. A third challenge will be to model the buffer requirement of OR protocols.

9.7 Chapter Summary

Performance modeling of opportunistic routing protocols is an essential part of research since the testing of new protocols is heavily dependent on simulation-based studies. Modeling complements the experimental results and enables a researcher to understand the intricacies and deficiencies of the system. In this chapter, we described the performance models available in literature. We found that these models assumed the protocol to be ideal in the sense that there were no duplicate packet transmissions. Thus the empirical results did not agree with the analytical result. Our analysis indicated that duplicate packet transmissions can occur if a lower priority forwarder cannot overhear the transmission of its following higher priority forwarder. A new mathematical model based on binomial distribution was introduced that accounts for such duplicate transmission. Based on the finding of our study, we proposed a opportunistic routing protocol where all consecutive forwarders are in the transmission range of one another. Empirical as well as analytical results show that such an approach lowers duplicate transmissions and as a results minimizes the total packet transmission. Finally, the chapter address the problems and challenges in modeling ad hoc opportunistic routing protocols.

References

1. Biswas SZ (2005) Opportunistic routing in multi-hop wireless networks. MS Thesis, MIT
2. Biswas S, Morris R (2004) Opportunistic routing in multi-hop wireless networks. ACM SIGCOMM Comput Commun Rev 34(1):69–74
3. Biswas S, Morris R (2005) ExOR: opportunistic multi-hop routing for wireless networks. SIGCOMM Comput Commun Rev 35(4):133–144
4. Chachulski S, Jennings M, Katti S, Katabi D (2007) Trading structure for randomness in wireless opportunistic routing. SIGCOMM Comput Commun Rev 37(4):169–180. doi:10.1145/1282427.1282400
5. Darehshoorzadeh A, Cerda-Alabern L, Pla V (2011) Modeling and comparison of candidate selection algorithms in opportunistic routing. Comput Netw 55(13):2886–2898
6. De Couto D, Aguayo D, Bicket J, Morris R (2003) A high-throughput path metric for multi-hop wireless routing. In: Proceedings of the ACM 9th annual international conference on mobile computing and networking (MobiCom ’03), New York, USA, pp 134–146. doi:10.1145/938985.939000
7. Durairaj V, Kalla P (2005) Enhanced interior gateway routing protocol, Cisco White Paper. In: Proceedings of the 8th international conference on theory and applications of satisfiability testing (SAT 2005), pp 415–422
8. Gray C, Santhapuri N, Nelakuditi S (2008) On Bit-Rate selection for opportunistic routing. In: Proceedings of the 5th IEEE annual communications society conference on sensor, mesh and Ad Hoc communications and networks workshops, 2008 (SECON Workshops ’08), 16–20 June 2008, pp 1–6. doi:10.1109/SAHCNW.2008.17
9. Johnson DB, Maltzand DA (1996) Dynamic source routing in ad hoc wireless networks. In: Imelinsky T, Korth H (eds) Mobile computing, Kluwer Academic Publishers, Dordrecht, p 153–181
10. Kim J, Ravindran B (2009) Opportunistic real-time routing in multi-hop wireless sensor networks. In: Proceedings of the ACM symposium on applied computing (SAC ’09), New York, USA, 8–12 March 2009, pp 2197–2201. doi:10.1145/1529282.1529766
11. Koutsonikolas D, Wang CC, Hu YC (2011) ECCACK: efficient network coding based opportunistic routing through cumulative coded acknowledgments. IEEE/ACM Trans Networking 19(5):1368–1381. doi:10.1109/TNET.2011.2111382
12. Kurkowski S, Camp T, Colagrosso M (2005) MANET simulation studies: the incredibles. SIGMOBILE Mob Comput Commun Rev 9(4):50–61, doi:10.1145/1096166.1096174
13. Li Y, Chen W, Zhang Z (2009) Optimal forwarder list selection in opportunistic routing. In: Proceedings of the IEEE 6th international conference on mobile Adhoc and sensor systems, (MASS ’09), 12–15 Oct 2009, pp 670–675. doi:10.1109/MOBHOC.2009.5336939

14. Mao X, Li X, Song W, Xu P, Moaveni-Nejad K (2008) Energy efficient opportunistic routing in wireless networks. In: Proceedings of the 12th ACM international conference on modeling, analysis and simulation of wireless and mobile systems (MSWiM '09), New York, USA, 26–29 Oct 2008, pp 253–260. doi:10.1145/1641804.1641847
15. Mazumdar AP, Sairam AS (2011) PBFS: a technique to select forwarders in opportunistic routing. In: Proceedings of the IEEE region 10 conference (TENCON'11) 21–24 Nov 2011, pp 149–153. doi:10.1109/TENCON.2011.6129081
16. OMNeT++ Network Simulation Framework (2012) http://www.omnetpp.org accessed on January, 2012
17. Perkins CE, Bhagwat P (1994) Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers. ACM SIGCOMM Comput Commun Rev 24(4):234–244. doi:10.1145/190809.190336
18. Subramanian AP, Buddhikot MM, Miller S (2006) Interference aware routing in multi-radio wireless mesh networks. In: Proceedings of the 2nd IEEE workshop on wireless mesh networks (WiMesh 2006), 25–28 Sept 2006, pp 55–63. doi:10.1109/WIMESH.2006.288620
19. Tsai T-C, Tsai S-T (2009) A cross-layer routing design for multi-interface wireless mesh networks. EURASIP J Wirel Commun Networking, Article 4, Feb 2009. doi:10.1155/2009/208524
20. Varga A, Hornig R (2008) An overview of the OMNeT++ simulation environment. In: Proceedings of the 1st international conference simulation tools and techniques for communications, networks and systems (Simutools '08), Belgium, pp 1–10
21. Westphal C (2006) Opportunistic routing in dynamic ad hoc networks: the OPRAH protocol. In: Proceedings of the IEEE international conference on mobile Adhoc and sensor systems, (MASS'06), Oct 2006, pp 570–573. doi:10.1109/MOBHOC.2006.278612
22. Xia F (2008) QoS challenges and opportunities in wireless sensor/actuator networks. Sensors 8(2):1099–1110
23. Zhong Z, Nelakuditi S (2007) On the efficacy of opportunistic routing. In: Proceedings of the 4th annual IEEE communications society conference on sensor, mesh and ad hoc communications and networks, (SECON '07), 18–21 June 2007, pp 441–450. doi:10.1109/SAHCN.2007.4292856
24. Zorzi M, Rao RR (2003) Geographic Random Forwarding (GeRaF) for ad hoc and sensor networks: multihop performance. IEEE Trans Mob Comput 2(4):337–348. doi:10.1109/TMC.2003.1255648

Chapter 10
Reliable Transport in Delay Tolerant Networks

Arshad Ali, Manoj Panda, Lucile Sassatelli, Tijani Chahed and Eitan Altman

Abstract In this paper, we provide a holistic picture of the research efforts towards the design and development of transport protocols for DTN environments. In the first part, we provide an exhaustive and insightful survey of the literature on transport protocols and proposals aimed at DTNs. In the second part, we describe a new reliable transport protocol based on coding. Our proposed protocol is targeted at terrestrial DTN environments consisting of a large number of highly mobile nodes with random mobility. The key idea behind our proposal is that the average dynamics under such a network setting can be captured by a fluid-limit model and the protocol parameters can be optimized based on the fluid-limit model. Through simplified versions of our proposal, we guide the readers in a step-by-step manner through the intricacies of obtaining deterministic fluid-limit models for networks where the dynamics can be stochastically modeled by a continuous time Markov chain with a large state space. We also provide the relevant background material so as to help the readers clearly understand the methodology and enable him/her to apply the technique to their own research problems.

A. Ali · M. Panda(✉) · T. Chahed
Telecom SudParis, 9 rue C. Fourier, 91011 Evry Cedex, France
e-mail: manoj.panda@it-sudparis.eu

A. Ali
e-mail: arshad.ali@it-sudparis.eu

T. Chahed
e-mail: tijani.chahed@it-sudparis.eu

L. Sassatelli
Laboratoire I3S, Université Nice Sophia Antipolis - CNRS, Sophia Antipolis, CNRS, France
e-mail: sassatelli@i3s.unice.fr

E.Altman
INRIA, 2004 Route des Lucioles, 06902 Sophia-Antipolis, France
e-mail: eitan.altman@sophia.inria.fr

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_10,

Keywords DTNs · Reliable transport · Random linear coding · Fluid-limit · Protocol parameter optimization

10.1 Introdcution

Delay Tolerant Networks (DTNs) are a class of networks characterized by intermittent connectivity and/or large transfer delays. In a DTN, at any given time, no contemporaneous end-to-end path may ever exist [16]. Examples of DTNs include sparse Mobile Ad hoc NETworks (MANETs). If the spatial density of mobile nodes in a MANET is low, then an end-to-end path between a source and a destination almost never exists, and two mobile nodes can communicate only when they come within the radio range of each other.

Apart from sparse MANETs, other examples of DTNs include sparse Vehicular Ad hoc NETworks (VANETs) [74], Inter-Planetary Networks (IPNs) [17], Pocket Switched Networks (PSNs) [45], Airborne Networks (ANs) [50], Mobile Social Networks (MSNs) [52] and UnderWater Networks (UWNs) [90]. IPNs are the *deep-space* application of DTNs. *Terrestrial* applications of DTNs include UWNs, PSNs, VANETs, ANs and networks for developing regions [73].

There can be several reasons for intermittent connectivity such as limited wireless radio range, sparsity of mobile nodes, limited energy resources, attack, and noise [47]. The large delays could be due to large propagation delays (as in deep-space DTNs) or due to intermittent connectivity itself.

Due to intermittent connectivity, the nodes adopt the *Store-Carry-and-Forward* paradigm of routing where a source has to depend on the mobility of other nodes, which act as “relays”, and data packets and ACKnowledgements (ACKs) get transferred between a source and a destination through one or more relays. Since, nodes exploit the opportunities of contacts with other nodes in order to forward their information, such forwarding strategy is sometimes also called “opportunistic routing” [12].

This paper is concerned with reliable transport in DTNs under opportunistic routing. The paper consists of two parts. The first part provides a survey of the transport layer issues (Sect. 10.2) and the existing transport protocols aimed at DTN environments (Sect. 10.3). The second part describes a class of reliable transport protocols developed by the authors (Sects. 10.4 and 10.5).

In the first part, we begin by summarizing the main issues that need to be addressed when designing protocols for DTN environments (Sect. 10.2). Then, we explain why traditional transport protocols like TCP, which has been developed for use in a wired Internet, are unsuitable in wireless networks, in general, and in DTNs, in particular (Sect. 10.3.1). The rest of the first part is devoted to an exhaustive and insightful survey of the existing transport protocols designed for DTNs. In particular, we divide the discussion into: (i) protocols that aim at deep-space DTNs (Sect. 10.3.2), (ii) protocols that aim at terrestrial DTNs (Sect. 10.3.3), and (iii) protocols that make use of coding to provide a *probabilistic* form of reliability with improved performance (Sect. 10.3.4).

In the second part, we describe a class of reliable transport protocols developed by us [5]. The proposed protocols are targeted at terrestrial environments consisting of a large number of highly mobile nodes with random mobility. The key idea behind our proposal is that the average dynamics under such a network setting can be captured by a fluid-limit model and the protocol parameters can be optimized based on the fluid-limit model. Our main proposal appeared in [5] and is described in Sect. 10.4. Analytical modeling and performance optimization of the network dynamics under our proposal is provided in Sects. 10.5.3 and 10.5.4, respectively. However, from a pedagogic point of view, we explain the method of deriving fluid-limit models through simplified versions of our main proposal (Sects. 10.5.1 and 10.5.2) along with the background on fluid-limit models (Appendices A and B) and formal derivation of the fluid-limit in the simplest case (Appendix C). The material in Sects. 10.5.1 and 10.5.2, and that in Appedices A, B, and C is entirely new, and has been written for the sole purpose of guiding the authors through the intricacies of deriving fluid-limit models. We also provide numerical results to validate our fluid-limit models and discuss the performance of our optimization procedures (Sect. 10.5.4).

10.2 Transport Layer Issues in Delay Tolerant Networks

DTNs can be divided into two main categories, namely, *deep-space* and *terrestrial*. The development of an efficient transport protocol for deep-space IPNs has been a major challenge to the research community. The deep-space networking paradigm poses some significant challenges such as:

- **Long propagation time**: This issue arises from long distances by which planets are separated, and poses critical challenges in the designing of suitable protocols. Depending on the orbital location of the planets, the end-to-end round trip time experienced in IPNs can range from minutes (Earth to Mars networks) to even hours (Earth-Jupiter networks) [2, 24].
- **High channel error rates**: The deep-space links usually have very high bit errors rates [24].
- **Bandwidth asymmetry**: The issue of high channel error rates is associated with high asymmetry in forward and return link bandwidth. The ratio of bandwidth of forward to reverse channels is typically on the order of 1000 : 1 in space missions [24].
- **Intermittent connectivity**: Networks involving Mars orbiters and landers and Earth stations not only encounter extremely high latency links but also the devices are not contactable for extended durations [37]. For example, almost no effective communication occurs for a number of weeks during the regular periods when Mars is near the Sun. A very common, though less extended, disruption occurs due to rain at an earth station [4]. This disruption can hinder scheduled communications [39] that may lead to the need for re-transmissions. Such re-transmissions may face huge wait time, i.e., until the next contact opportunity, which could be hours later.

The challenging characteristics of terrestrial DTNs have attracted the attention of the networking research community in the recent past, mainly due to a plethora of potential applications such as wildlife tracking and monitoring [51], connectivity in undeveloped regions [73], vehicular networks [53], mobile social networks [52] and pocket switched networks [45]. *Intermittent connectivity*, *short contact duration*, *high mobility*, *unknown mobility patterns*, *energy and storage exhaustion* are just a few of the potential issues that may be faced in a terrestrial DTN environment.

A reliable transport protocol deals with some fundamental issues, namely, delivery guarantee and confirmation, connection state management [79], congestion and flow control. Delivery guarantee through ACKs might incur excessive additional delay in DTNs due to bandwidth asymmetry. Use of "smart" ACK mechanisms such as Selective ACKs (SACKs) and/or Selective Negative ACKs (SNACKs) might be necessary. Long propagation delays combined with high bit error rates could pose serious problems in adjusting the sending rate (flow control) to quickly fill the bit-pipe and respond quickly to losses. Unlike wired networks where congestion occurs mainly due to randomness in traffic pattern, congestion in DTNs can occur due to unavailability of links for prolonged durations and/or frequent link failures and short contact durations/UP times. Unlike wired networks where losses mainly occur due to congestion, losses in DTNs could be due to high bit error rates and/or frequent link failures. In wired networks, the transport layer protocol usually aborts the connection due to disruption and Internet congestion control rules hold for intermediate routers. In DTNs, (i) connection may be kept alive despite disruption, and (ii) congestion control rules are reformulated as storage congestion control at DTN nodes.

10.3 Survey of Transport Protocols in DTNs

Much of the existing literature on DTNs focuses on the routing aspect and relatively fewer pieces of work deal with transport aspect. Fall and McCanne [30] discuss some important issues related to transport performance not specific to DTNs. The Delay Tolerant Network Research Group (DTNRG) [47] of the Internet Research Task Force (IRTF) has investigated security and transport layer issues, mostly in the direction of deep-space communications [34, 35, 71, 78, 80, 81]. The main focus of the DTNRG is on the Interplanetary DTN environment. Routing or transport layer issues for terrestrial DTNs have not received much attention from the DTNRG so far.

The remainder of this section is structured as follows. First, we explain why traditional transport protocols are not suitable for use in DTNs. In the next two sections, we survey the literature on transport protocols designed for DTNs, which can be classified into: (a) deep space, and (b) terrestrial. Then, we discuss some proposals that use coding to enhance reliability.

10.3.1 Unsuitability of Traditional Transport Protocols in DTNs

The User Datagram Protocol (UDP) [75] and the Transmission Control Protocol (TCP) [76] are the most widely used transport protocols in the Internet today. The former provides a connectionless service and it lacks delivery guarantee whereas the latter provides a connection-oriented service and ensures reliable data delivery with delivery confirmation. By far, TCP is the *de facto* standard for reliable data transmission.

In reliable data transfer, the data source wishes to ensure that all the information it sends arrives correctly and "in order" at the destination. In TCP, data packets are given unique sequence numbers so that the destination can detect which packet(s) is/are missing. To achieve reliability, the destination sends acknowledgements back to the source for each packet (or for each other packet, in case of delayed ACKs) received without errors. A packet i is considered to be lost if either it is not acknowledged within some time period, T_0, or if the ACKs for three packets sent more recently than packet i arrive at the source before the ACK for packet i (so called *triple duplicate ACKs* which triggers fast retransmission). In addition to this, TCP implements a flow/congestion control mechanism wherein packets are sent within a so called *congestion window*, whose size increases with the receipt of ACKs and decreases after a time-out or in case of triple duplicate ACKs.

TCP generally performs well in networks made up of links with low bit error rates. In networks with high bit error rates, such as those consisting of wireless links and mobile hosts, many of the assumptions made by TCP are violated, which results in end-to-end performance degradation in such networks. For example, TCP assumes that packet loss occurs due to network congestion and TCP adapts well to network congestion. However, it misinterprets packet losses due to corruption by interference and noise as being due to congestion. TCP's congestion avoidance mechanism can only make things worse in such situations. Moreover, TCP requires at least one stable path between the source and the destination to establish the connection used in data delivery. However, such a path may not exist in wireless networks. A detailed account of the limitations of TCP in wireless networks can be found in [63].

A solution to the problems mentioned above has been provided in [10] where the authors suggest to "signal" losses due to packet corruption (rather than congestion) from the link layer to the transport layer. Balakrishnan et al. [9] proposed a "snoop-TCP" protocol to improve TCP performance in wireless networks. Mahmoodi et al. [64] proposed a TCP-aware dynamic Automatic Repeat-reQuest (ARQ) algorithm, which is a cross-layer design that improves TCP's performance in wireless networks with link layer adaptation.

In addition to high bit error rates, MANETs might also suffer from link failures. TCP turns out to be very inefficient for reliable transport in MANETs, because it misinterprets losses due to link failures as losses due to congestion [44]. This is even worse in the case of DTNs which suffer from *frequent* and *prolonged* link failures [80]. Indeed, DTNs are usually sparse and the lack of connectivity between the source and the destination is common. Most of the time, it is impossible to establish a data connection and to confirm delivery [101]. Therefore, TCP is not functional in DTNs [31].

The performance of TCP depends upon the product of the transfer rate and the round-trip delay called the *bandwidth-delay* product [48]. High propagation delay, especially of satellite links [8], makes the arrival of ACKs slow and a long time is required for the transmission window to grow. TCP's closed loop transmission approach is proven to be not sufficient to tackle with propagation delays on the order of minutes, i.e., feedback comes back at the sender when the information included in the ACK(s) may already be useless. In other words, it is possible that actual congestion event happens several minutes before triggering of congestion control and recovery mechanisms by TCP's triple duplicate ACKs. Thus, TCP performance degrades under large bandwidth-delay product. Moreover, it is quite well known that TCP misinterprets losses due to link errors as losses due to congestion. The problem of improving TCP over satellite has been widely explored [6, 11, 38].

10.3.2 Transport Proposals for Deep-Space Communication

Space Internet or space Internetworking is to enable space communication using Internet-type protocols and the Internet in the deep-space environment is generally termed as Interplanetary Internet [3]. deep-space environment entails huge delays even in the presence of connectivity. The majority of the architectural designs and works, mainly [2, 19, 32, 36, 71, 81, 82, 86, 100], address deep-space communication.

The Transport Protocol for InterPlanetary Internet (TP-Planet) [2] is one of the early proposals for reliable data transmission over single deep-space links. Its main functionality is to probe congestion detection and control mechanism which deals with congestion losses. It uses a blackout state procedure to cope with blackouts and also makes use of delayed Selective ACK strategy to handle bandwidth asymmetry. To be more precise, assuming the presence of IP infrastructure in space, it has deployed rate-based Additive Increase Multiplicative Decrease (AIMD) congestion control. AIMD operation depends on the decision of the congestion detection mechanism. TP-Planet adapts AIMD parameters in order to compensate the throughput degradation due to propagation delay. The TP-Planet protocol has issued two new algorithms namely (i) the initial state and (ii) the steady state. The initial state algorithm can occupy the bandwidth of the links in a very fast and controlled manner. It helps in avoiding throughput degradation which occurs due to inefficient connection starting behavior. The steady state algorithm contains four states: (i) hold or monitoring rate (ii) increase or accelerating rate, (iii) decrease or slowing rate and (iv) blackout or interrupt state. At the start of the steady state where there is no change in transmission rate, the source goes to hold rate state. The TP-Planet deploys a new congestion control scheme during steady state operation which helps in deciding transitions between the states in the steady state. Hence, it can increase, decrease or hold data transmission rate according to the current state. On the intermittent connection, the steady state algorithm has introduced the program of interrupt. TP-Planet has brought in the time-delay option of SACK to deal with asymmetric bandwidth. The TP-Planet is an end-to-end connection, but the original node and the destination node may not exist in the end-to-end connection in the deep-space communication networks.

The RCP-Planet protocol [32] is an unreliable version of TP-Planet [2] proposed by the same authors. It employs a rate control mechanism which copes with link congestion and error rate as well as a packet-level Forward Error Correction (FEC). It deploys a blackout state procedure and FEC block-level ACKs in order to handle bandwidth asymmetry. Delivery of real-time application data to the ground or to the satellite, spacecraft etc is the main objective of RCP-Planet.

The Bundle Protocol (BP) [86] proposed by DTNRG [47] borrows from the concept of email protocol and describes the end-to-end exchange of messages called "bundles" in DTNs. The BP is proposed on top of the transport layer for deep space communication. The basic idea is to operate in store and forward mode. A bundle node consists of (i) a bundle protocol agent, (ii) a set of convergence layer adapters, and (iii) an application agent. The application agent makes a transmission request to initiate a bundle transmission. Selecting appropriate convergence layer adapters after the establishment of endpoints, an outbound bundle is created according to the request parameters. Then, this bundle is sent to all active nodes currently associated with the specified endpoint. A bundle receipt acknowledgement request can be made for each transmission. The BP agent performs bundle forwarding. It sends a bundle to all nodes currently associated with a specified endpoint which could be either (i) final destination endpoint or (ii) other intermediate endpoints with the ability to forward the bundle to the destination. Convergence layer adapters must notify the bundle protocol agent about the completion of the transfer. The custody transfer, a mechanism which ensures the storage/custody of a bundle copy in the forwarding node until the release of the custody, can be requested for the transaction of each bundle. Its has the ability to deal with intermittent connectivity through custody-based message transfers in store-and-forward mode and takes advantage of scheduled, predictive and opportunistic connectivity [94]. It addresses high BER by dividing the larger end-to-end path into smaller hop-by-hop transfer. The BP specifies a framework rather than a concrete protocol implementation. Besides, the current BP works only for the content whose size is already known before the transmission. The size of bundle may sometime increase considerably. As per [72], in case of disruptions or node failures, BP cannot assure application reliability due to the lack of an end-to-end functionality and thus does not really provide an end-to-end service as claimed in the Request For Comments (RFC) 5050 [86]. Wood et al. [99] highlights a set of issues and limitations of BP. In particular, reliability in BP's custody transfer function is prevented due to the lack of an error detection approach (e.g., checksum) specification [99].

Reliability through custody transfer in bundle layer protocol is briefly addressed in [29]. A custodian takes the responsibility of reliable delivery of the bundle to the next custodian till bundle is delivered to the destination.

The Space Communication Protocol Standards-Transport Protocol (SCPS-TP) [36] for space communications was proposed by the Consultative Committee for Space Data Systems (CCSDS). This protocol adopts major TCP functionalities and extends them in order to deal with some of the unique characteristics of deep-space links. This protocol has two modes of operations: (i) TCP-Vegas [13] type Van Jacobson congestion control mode and (ii) an open loop rate control. It operates in one of the two mentioned modes during congestion indications. By choosing the mechanism of the TCP-Vegas congestion control, it can adjust the window according to change of Round Trip Time (RTT). TCP-Vegas is able to monitor effectively the status of usage of network, which has been widely employed in the network of satellite for its better fairness and effectiveness. To address the limited bandwidth and to provide more efficient loss recovery, SCPS-TP makes use of header compression and Selective Negative ACKnowledgement (SNACK) options. Its open loop rate control mode also makes use of SNACKs instead of usual ACKs. The SCPS-TP can judge status of links weather broken or not by monitoring signal strength between the spacecraft and earth-based station. When the signal strength is lower than a threshold, the sender sends Internet Control Message Protocol (ICMP) to all the receiving stations till the links become normal again. Then in order to maintain the communication, it continues to send ICMP messages to readjust the rate of sending. To deal with bandwidth's asymmetry problem, the SCPS-TP adopts the time-delayed ACK approach.

The CCSDS File Delivery Protocol(CFDP) [19] has the capability to transfer files to and from a spacecraft's data storage. It is mainly an application layer protocol, but includes functionalities of transport layer too. The core procedures of CFDP provide file copy services over a single link, i.e., its operation is to copy files from a source storage medium to a target storage medium. It can offer both unreliable and reliable (based on SNACKs) services. In the absence of direct connectivity between source and destination nodes, the protocol introduces extended procedures capable of performing multiple file copy operations across each link of the path to the final destination. CFDP does not support route adaptation inherently, therefore, the complete autonomous hand-over operations like ground station hand-over are established by the system functions external to CFDP. CFDP includes four modes (deferred, immediate, prompted and asynchronous) for sending negative ACKs. It also makes use of positive ACKs to ensure the receipt of critical PDUs. The extended mode of operation of CFDP refers to more complicated situations, and provides store-and-forward feature across a network having multiple links with disparate availability. Current store-and-forward approach of CFDP requires that all parts of a file follow the same path from a source to a destination. It appears to be unsuitable with increased communication complexity like dynamic routing or file distribution through multiple relay points in parallel. A solution currently under [72] investigation is the integration of CFDP with BP [86].

For dedicated networks like deep-space with very high RTTs, the Licklider Transmission Protocol (LTP) [81] offers reliable transfers based on retransmission over single-hop connections. It inherits major design ideas from CFDP [19]. To support BP over long and/or frequent intermittent links, it serves as a DTN convergence layer protocol which is its major design purpose, though can be used in other contexts. Such links might be encountered in interplanetary network setting. LTP's standard use case is a single hop, deep-space link such as between a remote spacecraft and an earth station. It does not consider typical TCP issues such as flow and congestion control in shared networks. Thus, before triggering communication, it makes use of pre-defined parameters like window size in its current implementation. In a typically very high delay point-to-point environment operating over single links, LTP handles delay tolerance and disconnection issues. LTP data flows are unidirectional and do not perform any handshakes, flow or congestion control as compared to TCP [84]. LTP supports both reliable and unreliable data transmission. Since LTP is a point-to-point protocol, there are no routing or congestion issues to consider for LTP itself. Bytes are simply transferred between two peers without any consideration of intermediate peers. LTP can transfer unnamed data blocks and brings partial reliability by dividing each block into two parts, (i) reliable "red" part and (ii) unreliable "green" part. ACKs are sent only when explicit solicitations are encountered for checkpoints (reception reports) in the sequence of incoming data segments of the block's red part. It still allows deferred transmission in case the communication link is not available. LTP runs just above the link layer and may also be useful for some terrestrial applications like sensor networks [67].

Farrell and Cahill [33] proposed a transport protocol called LTP-T (a variant of LTP [81]) which targets deep-space applications. LTP-T uses the notion of custody transfer. Each LTP-T entity must accept custody for all successfully received blocks. Custody is thus passed from host to host until the final destination is reached. This can minimize end-to-end data delivery time, though the requirement of accepting custody for all received blocks can also lead to storage exhaustion problems. The protocol supports delay-tolerant transport and congestion notification. Reliability and related issues become more complex as compared to LTP since LTP-T is designed for multi-hop environment. Basically, LTP-T operates like a sequence of independent LTP sessions (one for each session) when no error occurs. In the presence of segment loss or corruption, successfully received segments are forwarded to the next node while error recovery is started for the others. This de-synchronization of the initial segment sequence brings in proper checkpoint re-scheduling. However, LTP-T is only defined in a generic way, and further details of how it would operate in reality are not known.

The Deep-Space Transport Protocol (DS-TP) [71] addresses efficient and reliable communication in deep-space. It favors missions with small connectivity time due to its ability to complete file transfer faster than conventional protocols like TCP. DS-TP adopts (i) open loop approach to deal with huge propagation delays, (ii) SNACKs (a common approach) to deal with bandwidth asymmetries in satellite and space communications, which are able to signal for multiple holes at the receiver's buffer in contrast to simple Negative ACKnowledgement (NACK), and (iii) fixed-rate based transmissions to allow for high link utilization without forcing increase in the transmission rate that ultimately leads to congestion losses. DS-TP focuses on the optimization of Double Automatic Retransmission (DAR), a retransmission approach of the transport protocol, to deal either with high bit error rates or blackouts and to provide proactive protection against link errors. DAR allows for fast and efficient "hole-filling" at receiver's buffer. The level of redundancy introduced by DS-TP affects both the storage space requirement of intermediate DTN nodes and the end-to-end delivery delay of data to its destination.

The Saratoga protocol [100] developed by [83] is a file transfer and content dissemination protocol. It was originally developed with the purpose to transfer large files from small low-Earth-orbit satellites, but is useful for many other situations including ad-hoc peer-to-peer communications and DTNs. Its intended use is for moving files or streaming data between peers with intermittent connectivity. Saratoga is capable of reliably transferring very large amount of data under adverse conditions. It provides an IP-based convergence layer in DTNs to exchange bundles between peer nodes, supporting store-and-forward of bundles. The store-and-forward delivery approach relies on reliable hop-by-hop transfers of files. Saratoga supports retransmission requests of missing segments, uncorrelated with the data sending process, and contains SNACKs in order to provide reliable retransmission of data. Based on scheduled connections, Saratoga also supports an optimistic data transfer mode, where the source, immediately after initiating a communication session, starts data transmission without waiting for an acknowledgement. There are at least two feasible options to transfer a larger file which cannot be transferred by Saratoga during a time-limited contact. These are: (1) proactive fragmentation that can be used by the application to create multiple smaller-sized files, enabling Saratoga to transfer some of these smaller files completely during a contact, and (2) use of HOLESTOFILL packet to avoid file fragmentation. The receiver can retain a partially-transferred file and request transfer of the unreceived parts during a later contact by using HOLESTOFILL packet in order to make clear that how much of the file has been successfully received and where transfer should be resumed from. The best case of Saratoga requires just two acknowledgements: one for session establishment and one after successful data receipt. Saratoga also supports an optimistic transfer mode which can be used in situations involving well known communicating parties and scheduled transactions. Generally, it can be stated that, although Saratoga may not always work so reliable, it is more flexible than TCP. It can be adapted to different communication scenarios, providing storage and forwarding of bundles. It provides efficient means for data transfer, minimizing overheads and maximizing throughput through the best exploitations of available resources.

The Delay-Tolerant Transport Protocol (DTTP) [82] was proposed to increase reliability and efficiency (in terms of resource utilization) in DTNs. To increase efficiency of space data transfer, the authors claim to introduce dynamic characteristics therein. For the same DTTP session, through potential multiple paths, it offers multi-hop data transfer services. Practically, it relies jointly on an open-loop model and a closed-loop system for transmission scheduling and administration respectively. DTTP provides both reliable and unreliable communication with trade-off between reliability and end-to-end delivery delay. To provide reliable communication, DTTP agents run on every node along multi-hop paths, and reliability is offered separately on each link of the path. In order to ensure complete transfer of data, it supports cumulative and selective ACKs used collaboratively in its ACK packets. Its custody transfer approach transfers reliable transfer responsibility from one node to the next on the path till final destination. Full reliability requires extensive retransmissions, which obviously extends overall transfer time.

10.3.3 Transport Proposals for Terrestrial DTNs

Terrestrial DTNs entail Internet like communication in terms of delays once the connectivity is there. Now, we detail a few proposals for terrestrial DTN environments [43, 70, 85, 87, 88].

The main function of the Persistent Connection Management Protocol (PCMP) [70] is to keep the TCP connection alive during disconnections in DTN environments like Drive-Thru. The idea of Drive-Thru Internet is to provide hot spots along the road within a city, on a highway, or even on high-speed freeways such as autobahns [69]. The approach of PCMP is similar to Snoop-TCP [9]. Important aspects related to data transfers within connectivity islands in Drive-Thru environments are addressed by PCMP. The connectivity of a car entering a road based WLAN is split into the entry, production and exit phases. In case the transfer does not complete during the production phase, then this issue is handled by keeping the state of the connection alive till the arrival of the car to the next connectivity island. The authors in their study [70], through real road experiment, claim transfer of big amounts of data even for high speed car. PCMP assumes that the next connectivity island for a car moving on a highway is known in advance opposed to real, wide-deployment cases where the movement of cars is normally arbitrary.

Schutz et al. [85] measure the behavior of Internet communication across a dynamically changing, intermittently connected path through experiments. Through analysis of experimental results, they observe that the address changes, transport-layer timeout and retransmission behavior (characteristics of intermittent connectivity) are the main limiting factors. On the basis of these results, they propose enhancement protocol by combing the Host Identity Protocol (HIP) [68] with the TCP User Timeout Option [25] and the TCP Retransmission Trigger [26]. Their enhanced protocol can avoid connection aborts due to disconnection periods, utilize more efficiently connectivity periods and avoid connection aborts due to IP address change by using TCP User Timeout Option, TCP Retransmission Trigger and HIP, respectively.

Harras and Almeroth in [43] elaborate on acknowledgement approaches for Mobile DTN (MDTN) environment. The scenario considered is as follows: a source node forwards a message to a destination node through relay nodes. The work considers four different acknowledgment strategies. Trade-offs between delivery reliability, queuing time and delivery ratio are investigated based on the acknowledgment strategies. The acknowledgment strategies are:

- In the Hop-by-hop approach, each relay node takes responsibility of custody transfer. Each relay, upon receiving the message successfully, sends back an ACK. ACKs are send along every hop in the path. The source as well as the relay nodes try to infect as many nodes as possible. The source assumes that the message will eventually reach the destination due to provision of enough time and mobility of nodes. Thus, hop-by-hop ensures some level of reliability, but it does not ensure end-to-end reliability.
- Next, in the active receipt approach, the destination generates an ACK (receipt) after receiving the message. The generated ACK actively flows back to the infected

nodes in order to cure them. Actually, nodes treat this ACK as a new message needed to be forwarded.

- To overcome the cost related to active receipt, the destination node sends a kill message (implicit/passive receipt) which travels back to the source. Relays carrying the kill message do not forward it, they simply wait for the active infected node to come in their communication range and they stop them sending the source message further.
- The network-bridged approach uses alternative technologies, like cellular network, in order to acknowledge the message receipt. It could make use of the cellular network, exploiting its availability, as an alternative path for communication, as the same could be used to transfer small control information due to its bandwidth limitations of transferring large amount of data. Thus, the cellular network functions as a bridge between MDTN nodes. This approach adds complexity of bridging the DTMN network with cellular network, the availability of the same cannot be guaranteed in a DTN setup.

Approaches to avoid storage congestion in DTNs are investigated by Seligman et al. [87, 88]. When a node becomes congested, it forwards some fraction of its stored messages to other nodes who take custody responsibility. Its storage routing can utilize an extended push-pull concept of custody transfer. Such forwarding decongests the node and makes it available to serve incoming traffic. The algorithm consists of two parts. In the first part, a node selects a message from the stored messages in order to forward the same. Then, it selects a node in its k-hop neighborhood to whom the message is to be forwarded. These works conclude that huge storage capacity might be required in order to achieve delivery reliability. Delivery delay may be increased potentially due to message circulation, i.e., data is exchanged between neighbor nodes and not forwarded towards its ultimate destination node. Buffer management issues in DTNs are also discussed by Krifa et al. [54, 55].

Bulut et al. [14] propose a multiperiod spraying approach for routing in DTNs and try to optimize the dissemination efficiency. The goal of the algorithm is to minimize the number of copies used while delivering a predefined percentage of all messages by a predefined delivery deadline. A time-dependent copying scheme is introduced for making copying decisions. This scheme actually considers the time remaining to the given delivery deadline. The algorithm partitions time from message creation to the predefined deadline into several variable-length periods consisting of a message spraying phase followed by a wait phase for message delivery. To switch from one period to another , they assume some “feedback”. However, the method of obtaining the feedback is not defined.

10.3.4 Proposals Based on Coding

Protocols based on rateless codes are appealing alternatives to TCP in a future Internet, in general, and in DTNs, in particular [7]. Assume that a file consisting of

K packets is to be transferred from a source to a destination. Protocols based on rateless codes allow the source and/or the relays to keep on sending newly generated packets, called Random Linear Combinations (RLCs), which are combinations of randomly selected packets that the nodes already have. The loss of an RLC can be compensated by another RLC, whereas without coding, the loss of a packet has to be compensated by retransmitting the same packet. With coding, the only thing that matters for the destination is to meet a certain number of Degrees of Freedom (DoFs) corresponding to the number of linear independent RLCs it has to receive in order to be able to decode the file, and not the receipt of specific packets.

For low-complexity rateless codes, such as Fountain and Raptor codes [60, 89], if the number K' of coded packets received at the destination is a little more than the initial number of information packets K, then with high probability, the destination can reconstruct the whole file. This success probability can be set as high as desired by properly choosing K'. For Random coding, or random linear network coding [62], K' can be equal to K to get high success probability, but the encoding and decoding complexities are, in this case, maximum.

Wang et al. [95] combined erasure coding with two-hop-relay [41], which splits and encodes the message into a set of smaller size blocks. Upon receiving a portion of these encoded blocks, the original message is reconstructed by the receiver. A hybrid routing approach provided in [20] is enhanced version of [95]. The approach in [20] generates a copy of each encoded block, performing transmission for both of them at each contact opportunity. The transfer of original block is similar to [95]. However, aggressive forwarding is used to transmit its copy during the remaining encounter duration soon after the first block is sent out. The use of a smaller coding rate provides the opportunity to split the message into many encoded blocks of smaller size. Though, reliable delivery is promoted in this way but more redundancy would be generated. On the other hand, a higher coding rate might not be sufficient for delivery. Inspired by this consideration, a proposal to adopt rateless codes instead of adopting to erasure coding is provided by Vellambi et al. [93].

For opportunistic networks, Lin et al. [58] demonstrated the performance of epidemic routing using network coding in comparison to the use of replication. Through simulations they showed that under bandwidth constraint, smaller block delivery delay can be achieved by using random linear coding than non- network coded packet forwarding. If buffer space within DTN nodes is limited, then they observe further improvement in delivery delay as compared to an uncoded dissemination.

Zhang et al. [102] investigated the use of random linear coding under epidemic routing for unicast applications in mobile DTNs. When bandwidth is constrained and packets are destined for the same destination from the same source, random linear coding over non-coded packets achieves smaller delay. This benefit further increases with limiting the buffer space in DTNs. With relatively loaded network, random linear coding achieves improvement over non-coding scheme only by appropriately controlling the spreading of information. Further, they observed that the benefit achieved by random linear coding for the *multiple sources-single destinations* case is smaller than for the *single source-single destination* case. They claimed that if there is a single block of packets in the network, for the single source/multiple sources-single destination, random linear coding achieves the minimum delay with high probability.

Bulut et al. [15], present multiperiod spraying algorithm for routing in DTNs. Erasure coding based technique is used to route message in order to ensure reliability against failures and increase the success rate of delivery. Their proposed scheme divides the original message into blocks which are individually transmitted. The destination decodes the original message by receiving only subset of original message or fewer blocks. Erasure based coding increases the chances of recovery of original message even if transmission failures occur. They address several cost reduction schemes maintaining the delivery rate and delay objectives. The number of periods is pre-decided. Other erasure coding based routing proposals are [95] and [96].

The authors in [57] combined network coding with epidemic routing [92] to attain a lower overhead ratio specially for dissemination of large number of messages.

Lin et al. [59] proposed E-NCP, an efficient protocol for DTNs. They combine network coding with binary spray-and-wait so as to control the number of transmissions significantly. Though, the reduction comes at the cost of slight increase in data transmission delay. However, using network coding makes the pre-determined maximum number of transmissions be more efficient for the delivery delay.

Jain et al. [49] considered the problem of routing in a DTN in the presence of path failures. They tried to address issues like how the copies should be distributed, or how many of them amongst the available paths. They presented a theoretical approach to determine set of paths to be used, with known and independent path failure probabilities. They formulated DTN routing as a resource allocation problem, with reliability as the goal.

Chen et al. [21] present a hybrid algorithm which integrates erasure coding with encounter prediction. They present message scheduling algorithms that enhance data delivery capabilities of the hybrid erasure coding scheme for file transfers. They propose a content-centric framework to facilitate data dissemination for video transfers and web surfing applications. Their proposed framework combines layered coding and multiple descriptions coding. The authors claim that their proposed schemes can achieve a much better latency performance for file transfers. CFP algorithm presented by Die et al. [23] is an optimal probabilistic forwarding scheme using fountain code to encode message and provide the forwarding rule. They model the probabilistic forwarding problem as an optimal stopping problem.

In [22], a coding-based forwarding protocol called vCF is proposed for vehicular ad hoc networks with scheduled routes such as bus systems. In vCF, each message is fragmented into a set of blocks. The blocks are encoded via linear network coding. Each node, upon seeing a contact opportunity schedules coded blocks to transmit, and drops the blocks when there is buffer overflow.

Widmer and Le Boudec [98] propose a communication algorithm based on network coding. Their proposed algorithm reduces the overhead of probabilistic routing algorithms. The authors claim through simulations that their algorithm achieves the reliability and robustness of flooding at a small fraction of the overhead. According to [18], the work in [98] is regarded as a hybrid of Gossip routing [42] and Epidemic routing [92].

A forwarding strategy called HubCode is proposed in [1]. Hubcode replicates the message only to the nodes deemed as being hub in the social based underlying topology of the DTN. The hubs use random linear network coding to encode multiple messages which are addressed to the same destination. Message are delivered to the destinations by using the hubs as relays. An extra exchange overhead occurs during the exchange of coefficient matrix required to check the linear independence.

10.4 A New Reliable Transport Protocol Based on Random Linear Coding

From the survey provided in the previous section, we find that protocols for traditional wired and wireless networks are unsuitable for DTNs. Existing solutions in MANETs rely on cross-layer signaling between the transport and lower layers so as to inform the former about route failures [9, 10, 64]. This strategy cannot be used in DTNs as only opportunistic routing can be performed. This motivates the development of new approaches to transport in DTNs. Most of the existing transport protocols for DTNs have been proposed primarily for deep-space communication. Also, Fall and Farrell [28] have reported that the preliminary architecture by the DTNRG meant for deep-space communication would fit poorly into terrestrial DTNs due to lack of design flexibility. Therefore, we focused on designing an efficient reliable transport protocol targeted mainly for terrestrial DTNs [5].

Our proposed protocol is targeted at terrestrial DTN environments consisting of a large number of highly mobile nodes with random mobility. The key idea behind our proposal is that the average dynamics under such a network setting can be captured by a fluid-limit model and the protocol parameters can be optimized based on the fluid-limit model. In the absence of feedback from the destination, the source cannot know for sure how many packets, be they coded or not, made it successfully to the destination, and hence, only a "probabilistic" form of reliability can be obtained; the source can only ensure a certain probability of successful delivery of packets to the destination. Use of end-to-end ACKs is essential to ensure that (coded or uncoded) packets have indeed reached the destination and bring in the "deterministic" form of reliability. Thus, our proposal makes use of both coding and ACKs. Coding improves the probability of successful delivery within given time budgets and ACKs provide the *deterministic* form of reliability.

We study a scheme in which transmission is organized in cycles. During one cycle, the source sends a given number of Random Linear Combinations (RLCs) corresponding to the number of Degrees of Freedoms (DoFs) still missing at the destination, in a back to back manner, without waiting for any feedback. Each RLC received at the destination triggers the sending back of a corresponding ACK, indicating to the source the new number of DoFs still missing at the destination. The cycle ends at the end of a time-out period, at which time, all RLCs are dropped in the relays. The process is repeated until the completion of the transmission of the whole file.

The remainder of this section is devoted to describing our proposal in detail.

10.4.1 Network Setting

To better explain our proposal, we focus on the transfer of packets from a single source to a single destination corresponding to a single *flow*. The network consists of $N_0 + 2$ mobile nodes. There is one source node, one destination node, and the remaining N_0 nodes act as relays for the considered flow. The source sends packets to the destination and the destination sends back ACKs confirming successful reception of the packets. Two nodes are said to "meet" when they come within the communication range of each other.

We assume that the relays have buffer capacity to store at most one packet or one ACK at any point in time. In reality, a relay could store multiple packets and ACKs. However, there would also be several other competing flows to share the buffer space at the relays. Thus, our assumption of buffer capacity of one packet or ACK can be viewed as saying that the buffer capacity of a relay is limited to one packet or ACK *per flow*.

To make room for other packets and ACKs of the same flow and/or packets and ACKs of other flows, a packet (of the flow under consideration) is retained in a relay buffer only for a duration τ_e, called the *buffer expiry time-out*, and then dropped. However, we assume that ACKs are never dropped to make room for other packets or ACKs, since ACKs are much smaller than packets and they contain valuable information. The duration τ_e depends on several factors such as the number of simultaneous flows, the buffer capacity at the relays etc. We view the buffer expiry time-out τ_e as a constraint imposed by such external factors, and for our purpose, *we assume that* τ_e *is a given network parameter*.

10.4.2 The Proposal

The objective is to transfer a file consisting of M information packets from the source to the destination in a reliable manner. To that end, we propose a variant of the packet transfer mechanism introduced in [61] by adapting to the DTN setting as follows.

Upon meeting with relays, the source generates new Random Linear Combinations (RLCs) of the M information packets, over a Galois field $\mathbb{F}_q$. The relay nodes do not perform further coding. Due to the opportunistic nature of communication in DTNs, packets and ACKs are replicated as in *epidemic routing* [91] in which each mobile keeps forwarding a copy of its packet (or ACK) to the other mobiles it meets, which also spread the packet (or the ACK) in that way. The details of packet and ACK replications is provided in Sect. 10.4.4.

To recover the M information packets, the destination needs to receive M Degrees of Freedom (DoFs), i.e., the rank of the matrix formed by the random coefficients of the received RLCs must be equal to M. With coding, the objective of the destination is to meet M DoFs by accumulating *any* M linearly independent RLCs, and not the reception of specific information packets. Accordingly, as in [61], the ACKs in our scheme indicate the number of DoFs missing at the destination, and not the receipt of specific information packets. The ACKs generated by the destination and received at the source provide the source with the necessary information to evaluate the progress of the transfer.

To ensure reliability in the presence of packet drops due to buffer expiry timeout, our scheme: (1) evaluates the progress of the transfer at appropriate intervals using the feedback information provided by the ACKs, and then (2) takes corrective actions accounting for the feedback information provided by the ACKs. Thus, the M information packets are transferred from the source to the destination in a reliable manner over multiple *cycles*.

For coding to help increase the throughput, it is necessary to send the RLCs faster than the ACKs come back. Hence, it is necessary to optimize the number of RLCs to be sent before considering an ACK, so as to maximize the throughput. Also, there could only be M different types of ACKs corresponding to the M missing DoFs, but the number of different RLCs could be much larger. Hence, there is no one-to-one correspondence between the ACKs and the RLCs, and the ACKs cannot override/kill the RLCs. Therefore, the ACKs have to compete with the RLCs for getting access to relay buffers. Since the relay buffers can be occupied to a significant degree by RLCs, by the time ACKs are generated, it is necessary to stop the spreading of RLCs and let the ACKs propagate back to the source. In view of the above arguments, we propose to optimize the number of RLCs to be sent, M_i, and two timers, namely, the *spreading time*, $\tau_{i,S}$ and the *waiting time*, $\tau_{i,W}$, depending on the missing DoF, i, at the beginning of each cycle. Note that the missing DoF, i, is from the point of view of the source indicated by the ACKs, which might be different from the actual DoF at the destination.

Our objective is to minimize the *mean transfer time* of the whole file, i.e., the time until receiving an ACK indicating 0 missing DoFs. The optimization is performed over the parameters $\{M_i, \tau_{i,S}, \tau_{i,W}\}$, $i = 1, \ldots, M$. Although we do not focus on this point in this work, it is worth noting that the parameters M_i, $\tau_{i,S}$ and $\tau_{i,W}$ can also be used to trade between delay and energy.

10.4.3 The Algorithm

Our scheme is detailed as follows:

- **Initialization**: $i \longleftarrow M$.
- **While** $i > 0$,
  - A new cycle begins with i missing DoFs (as viewed by the source). The source sends M_i RLCs back to back, where M_i is a function of i. Each time an empty relay meets the source, the source gives a new RLC to the relay until M_i RLCs have been sent.
  - Each RLC is spread for a duration $\tau_{i,S}$, called the *spreading time*, according to the replication scheme described in Sect. 10.4.4.

  - Each time a relay meets the destination, the destination sends an ACK informing the source how many DoFs are still needed to recover the M information packets. (The ACK generation and replication scheme is detailed in Sect. 10.4.4.)
  - After emitting the M_i-th RLC, the source waits for a duration $\tau_{i,S}$ to let the M_i-th RLC spread in the network, and then waits further for a duration $\tau_{i,W}$, called the *waiting time*. The purpose of the waiting time is to allow the ACKs to reach the source.
  - Replication of the RLCs stops during the ACK-wait phase. However, replication of the ACKs continues throughout the cycle. A copy of an RLC is retained in a relay buffer only for a duration τ_e, whereas a copy of an ACK is retained in the relay buffer throughout the cycle.
  - The cycle lasts for a total duration

    $$\tau_i := t_{M_i} + \tau_{i,S} + \tau_{i,W},$$

    where t_{M_i} denotes the time at which the M_i-th RLC is sent by the source.
  - At the end of the cycle: (i) all the relays drop the copy of the RLC or ACK they have, and (ii) the source considers the minimum of the missing DoFs indicated by all the ACKs it has received during the cycle. Let the minimum of the missing DoFs indicated by the ACKs be j.

- **Update**: $i \longleftarrow j$.
- **End While**.

10.4.4 RLC and ACK Replication

For the sake of brevity, we call a cycle, which begins with i missing DoFs, an i*-cycle*. For example, the first cycle begins with M missing DoFs, i.e., the first cycle is an M-cycle. In a i-cycle, the source sends M_i RLCs back to back. Each time an empty relay meets the source, the source gives a new RLC to the relay until M_i RLCs have been sent in the current i-cycle. Each of the M_i RLCs is spread by the source only once. The source makes use of the first M_i transmission opportunities to send M_i different RLCs.

In any i-cycle, we index the RLCs by k, $k = 1, 2, \ldots, M_i$, and index the ACKs by l, $l = 0, 1, \ldots, i-1$. Note that, *ACKl contains the information that the destination still needs l DoFs to recover the M information packets*. Let t_k denote the time at which RLC k is sent by the source. When a relay with a copy of RLC k meets with an empty relay during $(t_k, t_k + \tau_{i,S}]$, the empty relay gets a copy of RLC k. Thus, each RLC, after being sent by the source, is spread for a duration $\tau_{i,S}$, after which it is not replicated any more. Furthermore, a copy of an RLC is retained in a relay buffer only for a buffer time-out period of duration τ_e. An empty relay, which had an RLC (or an ACK) earlier but dropped the RLC (or the ACK) due to buffer time-out or cycle time-out, is also allowed to receive, carry and spread another RLC (or even the same RLC) or ACK subsequently, i.e., we allow *re-infection*. This is desirable from the point of view of implementation, since the relays do not have to remember the history of the RLCs and the ACKs they have already carried. When two nodes, which have different RLCs, meet, then there is no exchange.

When the destination receives an RLC, it updates the missing DoFs, generates an ACK indicating the missing DoF, and the RLC in the relay gets replaced with the latest ACK. When the destination is in a state with l missing DoFs, $l = 0, 1, \ldots, i - 1$, it gives ACK l to all the relays it meets, be they empty or not, except to those who already have ACK l. Note that, upon meeting an empty relay, the destination does not generate an ACK if it is in a state with i missing DoFs, i.e., if it has not received any useful RLC in the current i-cycle. When a relay with ACK l meets an empty relay, the empty relay gets a copy of ACK l. When a relay with ACK l meets another relay with ACK l', $l' > l$, then ACK l' is replaced by ACK l, since an ACK indicating a smaller number of missing DoFs provides more recent and more accurate information to the source. ACK 0 also replaces the RLCs, since ACK 0 indicates complete reception of the file and no more RLCs are required to reach the destination. Replication of the RLCs occur only during the RLC-spread phase, whereas replication of the ACKs continues throughout the cycle.

At the end of each cycle, the relays drop the copy of the RLC or ACK they have. This choice is to maintain the stability of the network as follows. Dropping the RLCs at the end of each cycle makes room for the RLCs and the ACKs of the same flow in subsequent cycles as well as for other flows. More importantly, the relays would not know when the transfer is complete. Thus, if the RLCs and the ACKs are not dropped at the end of each cycle, they would remain in the network, unnecessarily, for a very long time. The RLCs would ultimately be dropped due to buffer expiry, but the ACKs will not. Due to these practical considerations, we propose to drop the RLCs and the ACKs from the relay buffers at the end of each cycle.

10.4.5 Implementation Issues

The nodes can implement our reliable transport scheme without being time synchronous. First, the source and the destination must agree on the values of the number of information packets, M, and the coding field size, q, by a handshaking mechanism. This is similar to the "connection set-up" phase of TCP in which basic variables are exchanged and agreed upon. The cycle time-out τ_i and the spreading time $\tau_{i,S}$ are included in each RLC generated by the source, and are kept, as is, in every copy of the RLC. The buffer expiry time-out τ_e is generated afresh by each relay at the time of receiving a copy of the RLC, since τ_e is local to each relay. An RLC is spread for a duration $\tau_{i,S}$, and is dropped from the relay buffer at the earliest of the time-outs τ_i and τ_e. Since the destination generates ACKs only after receiving an RLC in the current cycle, the cycle time-out τ_i is copied into the destination's buffer and subsequently included in the ACKs as well. A final "connection release" mechanism is required in which the source informs the destination to clear all the variables corresponding to the flow under consideration. In the remainder of the work, we shall not discuss the connection set-up and release mechanisms any further.

10.5 Analytical Modeling and Performance Optimization

To optimize the parameters $\{M_i, \tau_{i,S}, \tau_{i,W}\}, i = 1, \ldots, M$, we first obtain an analytical model for the average network dynamics. Our model is based on the fluid-limit approach. A brief background on fluid-limit models has been provided in Appendices A and B. As explained in Appendices A and B, we consider a sequence of networks indexed by the number of relays, N, in the network. Our specific network has $N = N_0$ nodes, and hence, it would represent the N_0-th network in the sequence of networks we shall consider. We scale the number of sources and destinations as well as the pairwise meeting rates appropriately in the sequence of networks, let N to infinity in a certain sense, and obtain a fluid-limit model. The fluid-limit model provides the average network dynamics in terms of deterministic Ordinary Differential Equations (ODEs), which we apply to an optimization procedure and obtain optimal settings for the parameters $\{M_i, \tau_{i,S}, \tau_{i,W}\}, i = 1, \ldots, M$.

Before proceeding to obtain the fluid-limit model for the network dynamics under our proposed scheme, we first explain the methodology of obtaining fluid-limit models by considering a simpler scheme which involves neither coding nor buffer expiry. We begin by considering the transfer of a single packet in Sect. 10.5.1, and then generalize to the case of M packets in Sect. 10.5.2. Then, we obtain the fluid-limit model under our proposed scheme in Sect. 10.5.3. Finally, we describe our optimal procedure and discuss its performance in Sect. 10.5.4.

We assume that the successive inter-meeting times between any two specific relays, say i and j, $j \neq i$, are i.i.d. exponential random variables with parameter β_r. We assume that the successive inter-meeting times between the source (resp. the destination) and any specific relay are i.i.d. exponential random variables with parameter β_s (resp. β_d). We ignore the possibility that the destination may receive the packet directly from the source, which might occur in DTNs with negligible probability. Our assumption of i.i.d. exponential inter-meeting times is motivated by the works [40, 46, 103], in which the authors show via simulations that, for "random walk" and "random direction" mobility models, the assumption of i.i.d. exponential inter-meeting times provides extremely accurate approximations for actual inter-meeting times.

10.5.1 Single Packet Transfer Without Coding and Buffer Expiry

Suppose that the source wishes to transfer a single packet to the destination through the N_0 relays. Consider a cyclic scheme with a constant timeout period $T > 0$. At instants $0, T, 2T, 3T, \ldots$, the source begins spreading a new packet if the previous cycle was successful, i.e., if the source receives the ACK within T, or repeats the same packet otherwise. At the end of each cycle, all the nodes drop the copy of the packet or ACK they have. This choice of dropping at the end of the cycle is important since the relay nodes might not know if the source has indeed received the ACK. Consider the following replication method.

Packet Replication: When relay i, which is empty (i.e., relay i has neither a copy of the packet nor a copy of the ACK), meets with the source, relay i gets a copy of the packet. When relay j, which is empty, meets with another relay k, $j \neq k$, which has a copy of the packet, relay j gets a copy of the packet. When a relay, which has a copy of the packet, meets with the destination, the destination receives the packet.

ACK Replication: The destination sends an ACK for every received packet. The destination replaces the copy of the packet at the relay, which brings the packet to the destination, by an ACK. When relay i, which has a copy of the ACK, meets with another relay j, which is empty, relay j gets a copy of the ACK. When relay i, which has a copy of the ACK, meets with another relay j, which has a copy of the packet, the packet in relay j is replaced by the ACK. When a relay, which has a copy of the ACK meets with the source, the source receives the ACK.

Analytical Modeling: Consider a sequence of networks indexed by N, where the indexed N represents the number of relays in the N-th network. Let $n_s^{(N)}$ and $n_d^{(N)}$ denote the number of sources and destinations, respectively, in the N-th network. Our specific network contains $N = N_0$ relays with $n_s^{(N_0)} = 1$, $n_d^{(N_0)} = 1$. Each source (resp. destination) in the N-th network meets with a relay at rate $\beta_s^{(N)}$ (resp. $\beta_d^{(N)}$), and every pair of relays meet at rate $\beta_r^{(N)}$. Since, the network "resets" at the end of each cycle, we can model the evolution of the network in each cycle independent of other cycle. Thus, we can reset time to zero at the beginning of each cycle. Let $X^{(N)}(t)$ denote the number of relays that have a copy of the packet at time t. Let $Y^{(N)}(t)$, denote the number of relays that have a copy of the ACK at time t. The superscript N emphasizes the fact that the evolution of $X^{(N)}(t)$ and $Y^{(N)}(t)$ depends on the total number of relays, N, in the network. Under our mobility and packet replication models, $\{(X^{(N)}(t), Y^{(N)}(t)), t \geq 0\}$ is a Markov chain.

Our aim is to study the transient behavior of the above chain. When the number of relays, N, is small enough, a numerical approach is possible [46]. However, when N is large, say $N \geq 100$, an explicit characterization of the transient behavior is difficult due to the presence of non-linear and non-homogeneous transition rates. To obtain numerical results and gain insights when N is large, we analyze the network in the limit, as $N \to \infty$. To that end, as $N \to \infty$, we must have that [103].

1. The pairwise meeting rates $\beta_r^{(N)}$, $\beta_s^{(N)}$ and $\beta_d^{(N)}$ scale down with N such that the quantities $\lambda_r := N\beta_r^{(N)}$, $\lambda_s := N\beta_s^{(N)}$, and $\lambda_d := N\beta_d^{(N)}$, remain constant, independent of N,
2. The number of sources $n_s^{(N)}$ and the number of destinations $n_d^{(N)}$ scale up with N such that the ratios $s := n_s^{(N)}/N$ and $d := n_d^{(N)}/N$ remain constant, independent of N, and

3. The initial values $X^{(N)}(0)$ and $Y^{(N)}(0)$ scale up with N such that the ratios $x(0) := \frac{X^{(N)}(0)}{N}$ and $y(0) := \frac{Y^{(N)}(0)}{N}$ remain constant, independent of N, in each of the networks in the sequence of networks indexed by N.

The above *fluid-limit* scaling amounts to increasing the number of sources, destinations and relays to infinity by increasing the area of the network, but keeping their densities (in number per unit area) constant [103]. The number of sources, $n_s^{(N)}$, and the number of destinations, $n_d^{(N)}$, are both equal to 1 in our specific network; they, however, must scale up with N, as $N \to \infty$, so that the ratios $n_s^{(N)}/N$ and $n_d^{(N)}/N$ remain equal to their corresponding values in our specific network. Similarly, for the meeting rates. In summary, we set

$$s = \frac{n_s^{(N)}}{N} = \frac{1}{N_0}, d = \frac{n_d^{(N)}}{N} = \frac{1}{N_0}, \quad \text{and}$$

$$\lambda_r = N\beta_r^{(N)} = N_0\beta_r, \lambda_s = N\beta_s^{(N)} = N_0\beta_s, \lambda_d = N\beta_d^{(N)} = N_0\beta_d.$$

Applying Theorem 1 (see Appendix B), we observe that, for large N_0, $\mathsf{E}\big(X^{(N_0)}(t)\big)$ and $\mathsf{E}\big(Y^{(N_0)}(t)\big)$ are well-approximated by $N_0x(t)$ and $N_0y(t)$, respectively, where $x(t)$ and $y(t)$ are the unique solution of the Ordinary Differential Equations (ODEs), $\forall t, 0 < t < \infty$,

$$\begin{aligned} \frac{dx(t)}{dt} =& (s\lambda_s + \lambda_r x(t))(1 - x(t) - y(t)) - \lambda_r x(t)y(t) - d\lambda_d x(t) \\ \frac{dy(t)}{dt} =& d\lambda_d x(t) + \lambda_r y(t)(1 - x(t) - y(t)) + \lambda_r x(t)y(t) \end{aligned} \qquad (10.1)$$

with initial conditions $x(0) = \lim_{N\to\infty} \frac{X^{(N)}(0)}{N} = 0$ and $y(0) = \lim_{N\to\infty} \frac{Y^{(N)}(0)}{N} = 0$. A formal derivation of (10.1) can be found in Appendix C. Note that $\mathsf{E}\big(X^{(N_0)}(t)\big)$ and $\mathsf{E}\big(Y^{(N_0)}(t)\big)$ represent the expected values of the number of relays having a copy of the packet and the number of relays having a copy of the ACK, respectively, at time t in our specific network.

Delay Distributions: Let $P_X(t)$ denote the probability that the destination has received the packet by time t, and let $P_Y(t)$ denote the probability that the source has received the ACK by time t. For large N_0, the Cumulative Distribution Functions (CDFs) $P_X(t)$ and $P_Y(t)$ are well-approximated by the unique solution of the ODEs, $\forall t, 0 < t < \infty$, (see [5, 7])

$$\begin{aligned} \frac{dP_X(t)}{dt} =& \lambda_d x(t)(1 - P_X(t)) \\ \frac{dP_Y(t)}{dt} =& \lambda_s y(t)(1 - P_Y(t)) \end{aligned} \qquad (10.2)$$

with initial conditions $P_X(0) = 0$ and $P_Y(0) = 0$, where $x(t)$ and $y(t)$ are obtained by solving Eq. (10.1).

Performance Optimization: Since, the network "resets" at the end of each cycle, the network dynamics in each cycle can be modeled by Eqs. (10.1) and (10.2). The rate, $\theta_Y(T)$, at which packets are reliably transferred (with the source receiving back the ACK) under the cyclic scheme with timeout (or cycle duration) T is given by

$$\theta_Y(T) := \frac{P_Y(T)}{T}. \tag{10.3}$$

Note that $\theta_Y(T)$ can be interpreted as the *throughput* in packets/time and its inverse can be interpreted as the *mean delay* to transfer a single packet. The objective is to find the optimal cycle duration T, $T > 0$, to maximize $\theta_Y(T)$, which is equivalent to minimization of the mean transfer delay of a single packet.

10.5.2 Transfer of M Packets Without Coding and Buffer Expiry

Suppose now that the source wishes to send M packets to the destination through the N_0 relays. Let the packets be indexed by k, $k = 1, 2, \ldots, M$. The destination sends an ACK of type k for every received packet of type k.

Packet Replication: The source spreads each of the M packets with equal probability. When relay i, which is empty, meets with the source, relay i gets a copy of packet k, $k = 1, 2, \ldots, M$, with probability $1/M$. When relay i, which is empty, meets with another relay j, $j \neq i$, which has a copy of packet k, relay i gets a copy of packet k. When a relay, which has a copy of packet k, meets with the destination, the destination receives packet k.

ACK Replication: The relay which brings a copy of packet k to the destination replaces its copy of packet k by ACK k. When relay i, which has a copy of ACK k, meets with another relay j, which has a copy of packet k, packet k in relay j is replaced by ACK k. When relay i, which has a copy of ACK k, meets with another relay j, which is empty, relay j gets a copy of ACK k. When a relay, which has a copy of ACK k meets with the source, the source receives ACK k.

Analytical Model: We consider a sequence of networks indexed by the number of relays, N, as before. The number of sources and destinations, and the meeting rates are defined as before. Let $X_k^{(N)}(t)$ denote the number of relays that have a copy of packet k at time t. Let $Y_k^{(N)}(t)$ denote the number of relays that have a copy of ACK k at time t. By similar arguments as before, applying Theorem 1 (see Appendix B), we observe that, for large N_0, $\forall k$, $k = 1, 2, \ldots, M$, $\mathsf{E}\Big(X_k^{(N_0)}(t)\Big)$ and $\mathsf{E}\Big(Y_k^{(N_0)}(t)\Big)$ are well-approximated by $N_0 x_k(t)$ and $N_0 y_k(t)$, respectively, where $x_k(t)$ and $y_k(t)$ are obtained as the unique solution of the ODEs, $\forall t$, $0 < t < \infty$,

$$\begin{aligned}\frac{dx_k(t)}{dt} &= \left(s\lambda_s\left(\frac{1}{M}\right)+\lambda_r x_k(t)\right)(1-x(t)-y(t))-\lambda_r x_k(t)y_k(t)-d\lambda_d x_k(t)\\ \frac{dy_k(t)}{dt} &= d\lambda_d x_k(t)+\lambda_r x_k(t)y_k(t)+\lambda_r y_k(t)(1-x(t)-y(t))\end{aligned} \tag{10.4}$$

with initial conditions $x_k(0) = \lim_{N\to\infty}\frac{X_k^{(N)}(0)}{N} = 0$ and $y_k(0) = \lim_{N\to\infty}\frac{Y_k^{(N)}(0)}{N}$ $= 0$, and where $x(t) := \sum_{k=1}^{M} x_k$ and $y(t) := \sum_{k=1}^{M} y_k$.

Delay Distributions: Let, $\forall k$, $k = 1, 2, \ldots, M$, $P_{X_k}(t)$ denote the CDF of the delay random variable of packet k and let $P_{Y_k}(t)$ denote the CDF of the delay random variable of ACK k. As before, for large $N_0, \forall k, k = 1, 2\ldots, M$, the CDFs $P_{X_k}(t)$ and $P_{Y_k}(t)$ are well-approximated by the unique solutions of the ODEs, $\forall t, 0 < t < \infty$,

$$\begin{aligned}\frac{dP_{X_k}(t)}{dt} &= \lambda_d x_k(t)(1-P_{X_k}(t))\\ \frac{dP_{Y_k}(t)}{dt} &= \lambda_s y_k(t)(1-P_{Y_k}(t))\end{aligned} \tag{10.5}$$

with initial conditions $P_{X_k}(0) = 0$ and $P_{Y_k}(0) = 0$, where $x_k(t)$ and $y_k(t)$ are obtained by solving Eq. (10.4).

Performance Optimization: The performance metric, $\theta_Y^{(M)}(T)$, in this case, is the success rate of receiving ACKs for all the M packets within the timeout T, and is given by

$$\theta_Y^{(M)}(T) = \frac{\prod_{i=1}^{M} P_{Y_i}(T)}{T}. \tag{10.6}$$

Again, the objective is to find the optimal cycle duration T, $T > 0$, to maximize $\theta_Y^{(M)}(T)$, which is equivalent to minimization of the mean transfer delay of M packets.

10.5.3 Reliable Transport with Coding at Source and Buffer Expiry

We now apply the methodology described earlier to obtain the fluid-limit model pertaining to our proposed scheme. The material in this section is from [5].

Modeling the Network Dynamics During One Cycle: We model the dynamics of an i-cycle to study the spreading of the M_i RLCs and the i ACKs. We can model the network dynamics of any i-cycle independent of the earlier cycles, since the relays drop the copy of the RLC or ACK they have at the end of each cycle. Thus, we reset the time variable t to zero in the beginning of each cycle and study the network dynamics in an i- cycle for $t \in [0, \tau_i]$.

As before, we consider a sequence of networks indexed by the number of relays, N. Let $X_k^{(N)}(t)$, $k = 1, 2, \ldots, M_i$, denote the number of relays that have a copy of RLC k at time t. Let $Y_l^{(N)}(t)$, $l = 0, 1, \ldots, i-1$, denote the number of relays that have a copy of ACK l at time t. Let $P_{X_k}(t)$ denote the probability that the destination has received RLC k by time t, and let $P_{Y_l}(t)$ denote the probability that the source has received ACK l by time t. Let $Q_l^{(i)}(t)$ denote the probability that the destination has received $i-l$ DoFs in the current i-cycle by time t. Note that, $Q_l^{(i)}(t)$ also represents the probability that the number of missing DoFs at the destination at time t is l. Assuming that the reception of any RLC decreases the number of missing DoFs at the destination by one, $Q_l^{(i)}(t)$ is given by

$$Q_l^{(i)}(t) = \sum_{E\subset\{1,\ldots,M_i\}\,:\,|E|=i-l} \;\prod_{m\in E} P_{X_m}(t) \prod_{m'\in\{1,\ldots,M_i\}\setminus E} (1 - P_{X_{m'}}(t)). \quad (10.7)$$

In reality, reception of any RLC will not decrease the missing DoF by one, and hence, Eq. (10.7) is an approximation. Note, however, that *in the simulations, we actually check if a received RLC indeed decreases the missing DoFs.*

Applying Theorem 1 (see Appendix B), we observe that, for large N_0, $\forall k$, $k = 1, 2, \ldots, M_i$, $\forall l, l = 0, 1, \ldots, i-1$, the expectations $\mathsf{E}\Big(X_k^{(N_0)}(t)\Big)$ and $\mathsf{E}\Big(Y_l^{(N_0)}(t)\Big)$ are well-approximated by $N_0 x_k(t)$ and $N_0 y_l(t)$, respectively, where $x_k(t)$ and $y_l(t)$, are the unique solution of the ODEs

$$\frac{dx_k(t)}{dt} = \begin{cases} 0 \quad \text{for } 0 \le t \le \frac{k-1}{\lambda_s}, \\[2ex] (s\lambda_s + \lambda_r x_k(t))(1 - x(t) - y(t)) - d\lambda_d x_k(t) - \beta_e x_k(t) - \lambda_r x_k(t) y_0(t) \\ \text{for } \frac{k-1}{\lambda_s} < t \le \frac{k}{\lambda_s}, \\[2ex] \lambda_r x_k(t)(1 - x(t) - y(t)) - d\lambda_d x_k(t) - \beta_e x_k(t) - \lambda_r x_k(t) y_0(t) \\ \text{for } \frac{k}{\lambda_s} < t \le \frac{k}{\lambda_s} + \tau_{i,S}, \\[2ex] -d\lambda_d x_k(t) - \beta_e x_k(t) - \lambda_r x_k(t) y_0(t) \\ \text{for } \frac{k}{\lambda_s} + \tau_{i,S} < t \le \tau_i. \end{cases}$$

$$\frac{dy_l(t)}{dt} = \lambda_r y_l(t)(1 - x(t) - y(t)) + d\lambda_d Q_l^{(i)}(t)(1 - y_l(t)) + \lambda_r y_l(t) \sum_{m>l} y_m(t) \\ - \lambda_r y_l(t) \sum_{m<l} y_m(t) + \mathbf{1}_{\{l=0\}} \lambda_r y_l(t) x(t), \; \text{for } 0 < t \le \tau_i, \quad (10.8)$$

with initial conditions, $\forall k = 1, 2, \ldots, M_i$, $x_k(0) = 0$, and $\forall l = 0, 1, \ldots, i-1$, $y_l(0) = 0$, where $\lambda = N\beta$, $x(t) := \sum_{k=1}^{M_i} x_k$ and $y(t) := \sum_{l=1}^{i} y_l$. A detailed explanation of Eq. (10.8) can be found in [5].

It remains to obtain the $P_{X_k}(t)$'s and the $P_{Y_l}(t)$'s, which denote the probability that the destination has received RLC k and ACK l, respectively, by time t. As before, for large N_0, the CDFs $P_{X_k}(t)$, $k = 1, 2, \ldots, M_i$, and the CDFs $P_{Y_l}(t)$, $l = 0, 1, \ldots, i-1$, are well-approximated by the unique solution of the ODEs, $\forall t$,

$0 < t < \infty$, given by (10.5) with initial conditions $P_{X_k}(0) = 0$ and $P_{Y_l}(0) = 0$, where $x_k(t)$ and $y_l(t)$ are obtained by solving Eq. (10.8).

Combining the Cycles Together: We now proceed with the description of the sequence of cycles. Let Δ_n denote the number of DoFs missing at the destination in the beginning of the nth cycle. It is easy to see that, $\{\Delta_n, n \geq 1\}$ is a Markov chain with state space $\{0, 1, 2, \ldots, M\}$. The Markov chain $\{\Delta_n, n \geq 1\}$ begins with $\Delta_1 = M$, and gets absorbed in state 0. Let P_{ij} denote the transition probability from state i to state j.

As in [61], the transition probabilities can be expressed in terms of the *erasure probabilities* as seen by the source. In our context, given that a cycle begins with i missing DoFs (from the point of view of the source), the erasure probabilities correspond to $(1 - P_{Y_l}(\tau_i)), l = 0, 1, 2, \ldots, i-1$, i.e., the probability that the source has not received ACK l by the end of the i-cycle. Thus, the transition probabilities P_{ij}, $j = 0, 1, \ldots, i-1$, are given by

$$P_{ij} = P_{Y_j}(\tau_i) \prod_{l=0}^{j-1} (1 - P_{Y_l}(\tau_i)), \tag{10.9}$$

and

$$P_{ii} = 1 - \sum_{j=0}^{i-1} P_{ij}.$$

Let $T_i, i = 1, 2, \ldots, M$, denote the expected time to reach the state with 0 missing DoFs, starting from the beginning of an i-cycle. Clearly, T_M represents the expected completion time for the transfer of the whole file. By a renewal argument, we obtain, $\forall i, i = 1, 2, \ldots, M$,

$$T_i = \tau_i + \sum_{j=1}^{i} P_{ij} T_j, \tag{10.10}$$

which, in matrix form, can be written as

$$\boldsymbol{T} = \boldsymbol{\tau} + \boldsymbol{P}\boldsymbol{T} = (\boldsymbol{I} - \boldsymbol{P})^{-1}\boldsymbol{\tau} \tag{10.11}$$

where $T = (T_1, T_2, \ldots, T_M)'$, $\tau = (\tau_1, \tau_2, \ldots, \tau_M)'$,

$$\mathbf{P} = \begin{bmatrix} P_{11} & 0 & 0 & \ldots & 0 \\ P_{21} & P_{22} & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ P_{M1} & P_{M2} & P_{M3} & \ldots & P_{MM} \end{bmatrix},$$

I denotes the identity matrix (of appropriate dimension) and '$'$' represents the transpose.

Remark 10.1 The matrix $\boldsymbol{P}$ is obtained by deleting the row and the column corresponding to state 0 in the transition probability matrix of the Markov chain $\{\Delta_n, n \geq 1\}$. Hence, P is a sub-stochastic matrix and $\boldsymbol{I}$–$\boldsymbol{P}$ is invertible.

10.5.4 Optimization Procedure

Recall that our objective is to minimize the mean time to transfer the complete file, i.e., the time until receiving an ACK indicating 0 missing DoFs. For M packets, we need to minimize T_M over the parameters $\{M_i, \tau_{i,S}, \tau_{i,W}\}$, $i = 1, \ldots, M$. We now briefly describe the optimization procedure.

It can be seen from Eqs. (10.10) or (10.11) that the optimization can be recursive. Indeed, we have

$$T_i = \frac{\tau_i + \sum_{j=1}^{i-1} P_{ij} T_j}{1 - P_{ii}}.$$

Since T_i, $i = 2, \ldots, M$, depends only on $j = 1, \ldots, i-1$, we can perform a three-dimensional optimization (over the parameters $\{M_i, \tau_{i,S}, \tau_{i,W}\}$) at each step i, starting at $i = 1$, and then substituting T_j, for $j = 1, \ldots, i-1$, by the values obtained in the previous steps as

$$\min(T_i) = \frac{\tau_i + \sum_{j=1}^{i-1} P_{ij} \min(T_j)}{1 - P_{ii}}.$$

We thereby obtain the optimal values of $\{M_i, \tau_{i,S}, \tau_{i,W}\}$, $i = 1, \ldots, M$, in a recursive manner. For each step i, the three-dimensional optimization of the parameters $\{M_i, \tau_{i,S}, \tau_{i,W}\}$ pertains to the class of nonlinear optimization problems. Many general algorithms for solving such problems have been developed. We experimented with an algorithm called Differential Evolution (DE) [77]. DE is a robust optimizer for multivariate functions. We do not describe DE here, but only say that this algorithm is in part a hill climbing algorithm and in part a genetic algorithm.

The above optimization procedure does not have to be performed at the source node, but is rather performed offline, and the resulting optimal parameters for each possible i-cycle, $i = 1, 2, \ldots, M$, are stored in memory at the source node, so as to be used as needed.

In Fig. 10.1 we compare the mean file transfer time from simulations under the optimal settings with the file transfer time provided by the optimal procedure. It can be observed that the mean file transfer times from simulations, under the optimal settings of the parameters M_i, $\tau_{i,S}$ and $\tau_{i,W}$, are in excellent agreement with the optimal mean file transfer times. This validates our overall procedure of minimization of mean file transfer time based on our fluid-limit model. In Fig. 10.1, we also plot the mean cycle time, τ_M, staring with M missing DoFs. It can be seen that τ_M is close to the mean transfer time, T_M, which might involve multiple cycles. The closeness of T_M and τ_M

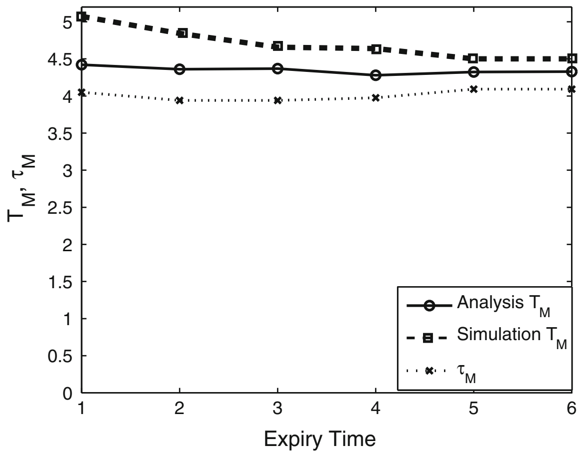


Fig. 10.1 Optimal values of the file transfer time T_M and comparison with simulation

suggests that our optimal settings of the parameters M_i, $\tau_{i,S}$ and $\tau_{i,W}$ are such that the transfer is mostly complete in just one cycle with high probability.

10.6 Conclusion

In this paper, we provide a holistic picture of the research efforts that have been carried out towards designing and developing transport protocols for DTN environments. Beginning with a summary of transport layer issues, we move on to explain why traditional transport protocols designed for a wired Internet cannot perform well in DTN environments. We then provided an exhaustive and insightful survey of the literature on transport protocols and proposals aimed at DTNs. In the second half of the paper, we described a new reliable transport protocol based on coding. Our proposed protocol is targeted at terrestrial DTN environments consisting of a large number of highly mobile nodes with random mobility. The key idea behind our proposal is that the average dynamics under such a network setting can be captured by a fluid-limit model and the protocol parameters can be optimized based on the fluid-limit model. Through simplified versions of our proposal, we guide the readers in a step-by-step manner through the intricacies of obtaining deterministic fluid-limit models for networks where the dynamics can be stochastically modeled by a continuous time Markov chain with a large state space. We also provide the relevant background material and the formal derivation of the fluid-limit model in the simplest case so as to help the readers clearly understand the methodology and enable him/her to apply the technique to their own research problems.

Several extensions to our proposal are worth pursuing. Handling the competition between several flows with relay buffers carrying multiple packets and ACKs appears to be the next step. Allowing coding and congestion control at the relays appears to be a challenging extension. The trade-off between delivery delay and energy is another important problem yet to be addressed.

Appendix A: Fluid-Limits and Fluid Approximations

In this appendix, we provide a brief background on fluid limits and fluid approximations. Please refer to [27, 56, 65, 66] and [97] for more details.

Intuitively speaking, a fluid-limit is a limit of a sequence of stochastic processes. The fluid approximation provides the first-order deterministic approximation to a stochastic process, and represents its *average* behavior.

Consider a sequence $\{Z^{(n)}(t), t \geq 0\}, n = 1, 2, \ldots,$ of stochastic processes. The index n represents some quantity which is *scaled up* to infinity in order to study the sequence of processes at the limit, as $n \uparrow \infty$. For queuing systems, n might represent "the number of servers" (as in infinite server approximations) or "a multiplying factor of one or more transition rates" (as in heavy-traffic approximations) or some other quantity w.r.t. which the scaling is performed. In our context of a mobile network, n might represent the number of mobile nodes. For a chemical reaction, n might represent the number of molecules and so on.

Recall that the Strong Law of Large Numbers (SLLN) (resp. Weak Law of Large Numbers (WLLN)) says that, under suitable conditions, the average of n random variables with a common mean μ, converges almost surely (resp. in probability) to μ, as $n \uparrow \infty$. Consider the SLLN (or WLLN) type rescaling $z^{(n)}(t) := Z^{(n)}(t)/n$. Under certain conditions, as $n \uparrow \infty$, the sequence of rescaled processes $\{z^{(n)}(t), t \geq 0\}$, $n = 1, 2, \ldots,$ converges almost surely (or sometimes in probability) to a deterministic process $\{z(t), t \geq 0\}$ (see, for example, Theorem 4.1 of [66]). Then, the limit $\{z(t), t \geq 0\}$ is called the *fluid limit* associated with the sequence $\{Z^{(n)}(t), t \geq 0\}$, $n = 1, 2, \ldots,$ and the approximation

$$Z^{(n)}(t) \approx n z(t) \ , \quad \forall t \geq 0 \ , \tag{10.12}$$

is called the *fluid approximation* for the nth system.

Appendix B: Density-Dependent Markov Chains

In this appendix, we recall a known fluid-limit result for the so-called *density dependent Markov chains*. First, we fix some notation. The set of integers (resp. real numbers) is denoted by $\mathbb{Z}$ (resp. $\mathbb{R}$). The space of d-dimensional vectors with integer (resp. real) components is denoted by $\mathbb{Z}^d$ (resp. $\mathbb{R}^d$). The absolute value of a scalar b is denoted by $|b|$. The norm of a vector z is denoted by $\|z\|$. The transpose of a vector z (resp. a matrix G) is denoted by z^T (resp. G^T).

Consider a one-parameter family of continuous time Markov chains $\{Z^{(N)}(t), t \geq 0\}$, indexed by $N = 1, 2, \ldots$, where $\{Z^{(N)}(t)\}$ has state space $\mathcal{S}^{(N)} \subset \mathbb{Z}^d$ and transition rate matrix $Q^{(N)} = [q^{(N)}(Z, Z')]$, $Z, Z' \in \mathcal{S}^{(N)}$.

Definition 10.1 (**Density Dependent Markov Chains [27, 56]**) The family of Markov chains $\{Z^{(N)}(t), t \geq 0\}$, $N = 1, 2, \ldots$, is called density-dependent if there exist a subset $\mathcal{R}$ of $\mathbb{R}^d$ and continuous functions f_l, $l \in \mathbb{Z}^d$, with $f_l : \mathcal{R} \to \mathbb{R}$, such that

$$q^{(N)}(Z, Z + l) = N f_l\left(\frac{Z}{N}\right), \qquad l \neq 0.$$

In practice, instead of considering all possible $l \in \mathbb{Z}^d$, one only needs to consider the much smaller set

$$\mathcal{L} = \{l \in \mathbb{Z}^d : l \neq 0, q^{(N)}(Z, Z + l) \neq 0 \text{ for some } Z \in \mathcal{S}^{(N)}\},$$

whose elements correspond to (actual) transitions of positive rate. For all $l \notin \mathcal{L}$, one can set f_l identically equal to zero. Henceforth, we only consider transitions with positive rates and denote the total number of such transitions by $|\mathcal{L}|$.

Define the *drift function* $F(\cdot)$ by

$$F(w) := \sum_l l f_l(w), \quad w \in \mathcal{R}. \tag{10.13}$$

Note that

$$F(w) = (F_1(w), \ldots, F_d(w))^T$$

is a d-dimensional column vector of functions, because l is a d-dimensional column vector. Defining the *density process* $\{z^{(N)}(\cdot)\}$ by

$$z^{(N)}(t) = \frac{Z^{(N)}(t)}{N},$$

we recall the Functional Strong Law of Large Numbers (FSLLN) for density-dependent Markov chains (see [Chap. 11, Theorem 2.1][27]).

Theorem 10.1 *(***Kurtz [27]***) Suppose that for each compact set $K \subset \mathcal{R}$,*

$$\sum_l \|l\| \sup_{w \in K} f_l(w) < \infty,$$

and there exists $M_K > 0$ such that

$$\|F(w) - F(w')\| \leq M_K \|w - w'\|, \quad \forall w, w' \in K.$$

Suppose also that

$$\lim_{N\to\infty} z^{(N)}(0) = z_0,$$

and $z(\cdot)$ *satisfies*

$$z(t) = z_0 + \int_0^t F(z(u))du, \quad t \geq 0. \tag{10.14}$$

Then, for every t, $0 \leq t < \infty$,

$$\lim_{N\to\infty} \sup_{0\leq s\leq t} \|z^{(N)}(s) - z(s)| = 0,$$

almost surely.

Remark 10.2 Theorem 1 says that, when the drift function $F(\cdot)$ is uniformly bounded and Lipschitz continuous over compact subsets of $\mathcal{R}$, then, as $N \to \infty$, the density process $\{z^{(N)}\}$ converges uniformly over compact subsets (u.o.c.) to a deterministic function $z(\cdot)$, almost surely (a.s.).

Appendix C: Derivation of Eq. (10.1)

In this appendix, we provide justifications for the ODE models which we have used as approximations for modeling the network dynamics when the number of relays N is large. We formally derive only Eq. (10.1). Derivation of fluid-limit models is similar for the other cases.

In the case of single packet transfer without buffer expiry, the positive transition rates of the Markov chain $\{(X^{(N)}(t), Y^{(N)}(t)), t \geq 0\}$ out of the state (X, Y) are given by

$$\begin{pmatrix} X \\ Y \end{pmatrix} \longrightarrow \begin{pmatrix} X+1 \\ Y \end{pmatrix} \quad \text{at rate} \quad \left(\beta_s^{(N)} n_s^{(N)} + \beta_r^{(N)} X\right)(N - X - Y)$$

$$\begin{pmatrix} X \\ Y \end{pmatrix} \longrightarrow \begin{pmatrix} X \\ Y+1 \end{pmatrix} \quad \text{at rate} \quad \beta_r^{(N)} Y(N - X - Y)$$

$$\begin{pmatrix} X \\ Y \end{pmatrix} \longrightarrow \begin{pmatrix} X-1 \\ Y+1 \end{pmatrix} \quad \text{at rate} \quad \beta_d^{(N)} n_d^{(N)} X + \beta_r^{(N)} XY$$

Writing $Z = (X, Y)$, the positive transition rates out of the state (X, Y) of the Markov chain $\{(X^{(N)}(t), Y^{(N)}(t)), t \geq 0\}$ can be written in the density-dependent form $N f_l\left(\frac{Z}{N}\right)$ given by (see Appendix B for a meaning of the density-dependent form)

$$Nf_l\left(\frac{Z}{N}\right) = \begin{cases} N\left(\left((N\beta_s^{(N)})\left(\frac{n_s^{(N)}}{N}\right) + (N\beta_r^{(N)})\left(\frac{X}{N}\right)\right)\left(1 - \frac{X}{N} - \frac{Y}{N}\right)\right) \\ \text{for} \quad l = (1, 0)^T, \\ N\left((N\beta_r^{(N)})\left(\frac{Y}{N}\right)\left(1 - \frac{X}{N} - \frac{Y}{N}\right)\right), \quad \text{for} \quad l = (0, 1)^T, \\ N\left((N\beta_d^{(N)})\left(\frac{n_d^{(N)}}{N}\right)\left(\frac{X}{N}\right) + (N\beta_r^{(N)})\left(\frac{X}{N}\right)\left(\frac{Y}{N}\right)\right), \\ \text{for} \quad l = (-1, 1)^T, \end{cases}$$

provided that $\lambda_r := N\beta_r^{(N)}$, $\lambda_s := N\beta_s^{(N)}$, $\lambda_d := N\beta_d^{(N)}$, $s := n_s^{(N)}/N$ and $d := n_d^{(N)}/N$ are constants, independent of N (which we assume, as in [103]). Let $z = (x, y)$. Defining $f_l(z) = f_l(x, y)$ and $F(z) = F(x, y)$ by

$$f_l(x, y) = \begin{cases} (s\lambda_s + \lambda_r x)(1 - x - y), & l = (1, 0)^T, \\ \lambda_r y(1 - x - y), & l = (0, 1)^T, \\ d\lambda_d x + \lambda_r xy, & l = (-1, 1)^T, \end{cases}$$

and

$$F(x, y) = \sum_l lf(x, y, w) = \begin{bmatrix} (s\lambda_s + \lambda_r x)(1 - x - y) - \lambda_r xy - d\lambda_d x \\ d\lambda_d x + \lambda_r y(1 - x - y) + \lambda_r xy \end{bmatrix},$$

respectively, and applying Theorem 1 (see Appendix B), we observe that, as $N \to \infty$, the rescaled process $\left\{\left(\frac{X^{(N)}(t)}{N}, \frac{Y^{(N)}(t)}{N}\right), t \geq 0\right\}$ converges almost surely to the unique solution of the ODE

$$\frac{dz(t)}{dt} = F(z(t)),$$

i.e., to the unique solutions $x(t)$ and $y(t)$ of the ODEs given by Eq. (10.1), with initial conditions $x(0) := \lim\limits_{N\to\infty} \dfrac{X(0)}{N} = 0$ and $y(0) := \lim\limits_{N\to\infty} \dfrac{Y(0)}{N} = 0$.

References

1. Ahmed S, Kanhere S (2009) Hubcode: message forwarding using hub-based network coding in delay tolerant networks. In: Proceedings of the 12th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems, pp 288–296
2. Akan OB, Fang J, Akyildiz IF (2004) TP-Planet: a reliable transport protocol for InterPlanetary internet. IEEE J Sel Areas Commun (JSAC) 22(2):348–361
3. Akyildiz I, Akan Ö, Chen C, Fang J, Su W (2003) Interplanetary internet: state-of-the-art and research challenges. Comput Netw 43(2):75–112

4. Albee A, Palluconi F, Arvidson R (1998) Mars global surveyor mission: overview and status. Science 279(5357):1671–1672
5. Ali A, Altman E, Chahed T, Panda M, Sassatelli L (2011) A new reliable transport scheme in delay tolerant networks based on acknowledgments and random linear coding. In: IEEE 2011 23rd International Teletraffic Congress (ITC), pp 214–221
6. Allman M, Dawkins S, Glover D, Griner J, Tran D, Henderson T, Heidemann J, Touch J, Kruse H, Ostermann S et al (2000) Ongoing tcp research related to satellites. RFC2760
7. Altman E, Pellegrini FD, Sassatelli L (2010) Dynamic control of coding in delay tolerant networks. In: Proceedings of IEEE Infocom, pp 1–5
8. Balakrishnan H, Padmanabhan V, Katz R (1999) The effects of asymmetry on tcp performance. Mob Netw Appl 4(3):219–241
9. Balakrishnan H, Seshan S, Amir E, Katz R (1995) Improving TCP/IP performance over wireless networks. In: Proceedings of the 1st ACM annual international conference on mobile computing and networking, pp 2–11
10. Balakrishnan H et al (1998) Challenges to reliable data transport over heterogeneous wireless networks. University of California, Berkeley
11. Barakat C, Altman E, Dabbous W (2000) On tcp performance in a heterogeneous network: a survey. IEEE Commun Mag 38(1):40–46
12. Biswas S, Morris R (2005) ExOR: Opportunistic multi-hop routing for wireless networks. In: Proceedings of Sigcomm, pp 133–144
13. Brakmo L, Peterson L (1995) TCP Vegas: end to end congestion avoidance on a global internet. IEEE J Sele Areas Commun 13(8):1465–1480
14. Bulut E, Wang Z, Szymanski B (2010) Cost-effective multiperiod spraying for routing in delay-tolerant networks. IEEE/ACM Trans Netw (TON) 18(5):1530–1543
15. Bulut E, Wang Z, Szymanski B (2010) Cost efficient erasure coding based routing in delay tolerant networks. In: 2010 IEEE international conference on communications (ICC), pp 1–5
16. Burleigh S, Hooke A, Torgerson L, Fall K, Cerf V, Durst B, Scott K, Weiss H (2003) Delay-tolerant networking: an approach to interplanetary internet. IEEE Commun Mag 41(6):128–136
17. Caini C, Cruickshank H, Farrell S, Marchese M (2011) Delay-and disruption-tolerant networking (DTN): an alternative solution for future satellite networking applications. Proc IEEE 99:1–18
18. Cao Y, Sun Z Routing in delay/disruption tolerant networks: a taxonomy, survey and challenges
19. CCSDS: CCSDS File Delivery Protocol (CFDP). In: CCSDS 727.0-B-4, Blue Book (2007)
20. Chen L, Yu C, Sun T, Chen Y, Chu H (2006) A hybrid routing approach for opportunistic networks. In: Proceedings of the 2006 ACM SIGCOMM workshop on Challenged networks, pp 213–220, Pise, Italy, Sept. 11–15
21. Chen L, Yu C, Tseng C, Chu H, Chou C (2008) A content-centric framework for effective data dissemination in opportunistic networks. IEEE J Sel Areas Commun 26(5):761–772
22. Chung KC, Li YC, Liao W (2010) Exploiting network coding for data forwarding in delay tolerant networks. In: 2010 IEEE 71st Vehicular Technology Conference (VTC 2010-Spring), pp 1–5
23. Dai Y, Yang P, Chen G, Wu J (2010) Cfp: Integration of fountain codes and optimal probabilistic forwarding in dtns. In: 2010 IEEE global telecommunications conference (GLOBECOM 2010), pp 1–5
24. Durst R, Feighery P, Scott K (2000) Why not use the standard internet suite for the interplanetary internet? In: Interplanetary internet study seminar, California Institute of Technology-1999
25. Eggert L, Gont F (2009) Tcp user timeout option
26. Eggert L, Schütz S, Schmid S (2005) Tcp extensions for immediate retransmissions. draft-eggert-tcpm-tcp-retransmit-now-02 (work in progress)
27. Ethier S, Kurtz TG (2005) Markov processes: characterization and convergence. Wiley Series in Probability And Statistics, Wiley Interscience, Published Online: 27 May 2008, ISBN: 9780470316658, doi:10.1002/9780470316658

28. Fall K, Farrell S (2008) DTN: an architectural retrospective. IEEE J Sel Areas Commun 26(5):828–836
29. Fall K, Hong W, Madden S (2003) Custody transfer for reliable delivery in delay tolerant, networks. IRB-TR-03-030
30. Fall K, McCanne S (2005) You don't know jack about network performance. Queue 3(4):54–59
31. Fall K (2003) 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, ACM, New York, pp 27–34, http://doi.acm.org/10.1145/863955.863960
32. Fang J, Akyildiz IF (2007) RCP-Planet: a rate control protocol for interplanetary internet. Int J Satell Commun Netw 25(2):167–194, http://dx.doi.org/10.1002/sat.873
33. Farrell S, Cahill V (2005) LTP-T: a generic delay tolerant transport protocol. Technical Report TCD-CS-2005-69, Computer Science, Trinity College Dublin
34. Farrell S, Cahill V (2007) Evaluating LTP-T: A DTN-Friendly transport protocol. In: 2007 IWSSC '07 International Workshop onSatellite and space communications, pp 178–181
35. Farrell S, Ramadas M, Burleigh S (2008) Licklider transmission protocol-security extensions
36. For Space Data Systems, C.C. (2006) Space Communications Protocol Standards (SCPS) - Transport Protocol (SCPS-TP). In: CCSDS 714.0-B-2, Blue Book
37. Franklin S, Slonski J, Kerridge S, Noreena G, Townes S, Schwartzbaum E, Synnott S, Deutsch M, Edwards C, Devereaux A et al (2004) The 2009 mars telecom orbiter mission. In: IEEE Proc of IEEE on aerospace conference, vol 1, 6-13 March 2004, Big Sky, MT, USA
38. Ghani N, Dixit S (1999) Tcp/ip enhancements for satellite networks. IEEE Commun Mag 37(7):64–72
39. Graf J, Zurek R, Eisen H, Jai B, Johnston M, DePaula R (2005) The mars reconnaissance orbiter mission. Acta Astronaut 57(2):566–578
40. Groenvelt R, Nain P, Koole G (2005) The message delay in mobile Ad Hoc networks. Perform Eval 62:210–228
41. Grossglauser M, Tse D (2002) Mobility increases the capacity of ad hoc wireless networks. IEEE/ACM Trans Netw 10(4):477–486
42. Haas Z, Halpern J, Li L (2002) Gossip-based ad hoc routing. In: Proceedings of IEEE, INFOCOM 2002. Twenty-first annual joint conference of the IEEE computer and communications societies. vol. 3, pp 1707–1716
43. Harras KA, Almeroth KC (2006) Transport Layer Issues in delay tolerant mobile Networks. In: IFIP Networking, Coimbra, Portugal
44. Holland G, Vaidya N (2002) Analysis of TCP performance over mobile ad hoc networks. ACM Wirel Netw 8(2):275–288
45. Hui P, Chaintreau A, Scott J, Gass R, Crowcroft J, Diot C (2005) Pocket switched networks and human mobility in conference environments. In: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking, pp 244–251
46. Ibrahim M, Hanbali AA, Nain P (2007) Delay and resource analysis in MANETs in presence of throwboxes. Perform Eval 24(9–12):933–945
47. Internet Research Task Force Delay-Tolerant Networking Research Group, http://www.dtnrg.org
48. Jacobson V, Braden R, Borman D (1992) Tcp extensions for high performance, http://coders.meta.net.nz/~perry/rfc/index-1323.html (Last retrieved April 23, 2013)
49. Jain S, Demmer M, Patra R, Fall K (2005) Using redundancy to cope with failures in a delay tolerant network. In: ACM SIGCOMM computer communication review. vol. 35, pp 109–120
50. Jonson T, Pezeshki J, Chao V, Smith K, Fazio J (2008) Application of delay tolerant networking (DTN) in airborne networks. In: Military communications conference, 2008. MILCOM 2008, IEEE, pp 1–7
51. Juang P, Oki H, Wang Y, Martonosi M, Peh L, Rubenstein D (2002) Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet. In: ACM Sigplan Notices. vol 37, ACM, pp 96–107

52. Kayastha N, Niyato D, Wang P, Hossain E (2011) Applications, architectures, and protocol design issues for mobile social networks: a survey. Proc IEEE 99(12):2130–2158
53. Khabbaz M, Fawaz W, Assi C (2011) Probabilistic bundle relaying schemes in two-hop vehicular delay tolerant networks. Commun Lett IEEE 15(3):281–283
54. Krifa A, Barakat C, Spyropoulos T (2011) Mobitrade: trading content in disruption tolerant networks. In: Proceedings of the 6th ACM workshop on Challenged networks, ACM, pp 31–36
55. Krifa A, Barakat C, Spyropoulos T (2012) Message drop and scheduling in DTNs: theory and practice. IEEE Trans Mobile Comput 11(9):1470–1483
56. Kurtz TG (1970) Solutions of ordinary differential equations as limits of pure jump markov processes. J Appl Probab 7(1):49–58
57. Lin Y, Liang B, Li B (2007) Performance modeling of network coding in epidemic routing. In: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking, ACM, pp 67–74
58. Lin Y, Li B, Liang B (2008) Stochastic analysis of network coding in epidemicRouting. IEEE J Sel Area Comm 26(5):794–808
59. Lin Y, Li B, Liang B (2008) Efficient network coded data transmissions in disruption tolerant networks, pp 1508–1516
60. Luby M (2002) LT codes. In: IEEE FOCS, pp 271–282
61. Lucani D, Stojanovic M, Médard M (2009) Random linear network coding for time division duplexing: when to stop talking and start listening. In: Proc. of IEEE Infocom 2009, pp. 1800–1808 Rio de Janeiro, Brazil
62. Lun DS, Médard M, Effros M (2004) On coding for reliable communication over packet networks. In: Proceedings of 42nd annual allerton conference on communication, control, and, computing, pp 20–29
63. Macedo D, dos Santos A, Pujolle G (2008) From tcp/ip to convergent networks: challenges and taxonomy. Commun Surv Tutorials IEEE 10(4):40–55
64. Mahmoodi T, Friderikos V, Holland O, Hamid Aghvami A (2007) Cross-layer design to improve wireless tcp performance with link-layer adaptation. In: Vehicular technology conference, 2007. VTC-2007 Fall. 2007 IEEE 66th, IEEE, pp 1504–1508
65. Mandelbaum A, Massey W, Reiman MI (1998) Strong approximations for markovian service networks. Queueing Syst 30:149–201
66. Mandelbaum A, Pats G (1995) State-dependent queues: approximations and applications. In: Kelly F, Williams RJ (eds) IMA volumes in mathematics and its applications, vol 71. Springer, Berlin, pp 239–282
67. Muhammad F, Franck L, Farrell S (2007) Transmission protocols for challenging networks: Ltp and ltp-t. In: International workshop on satellite and space Ccommunications, 2007. IWSSC'07, IEEE, pp 145–149
68. Nikander P, Moskowitz R (2006) Host identity protocol (hip) architecture. RFC 4423, http://www.ietf.org/rfc/rfc4423.txt (Last visited April 23, 2013)
69. Ott J, Kutscher D (2004) Drive-thru internet: Ieee 802.11 b for automobile users.In:Proc. of IEEE Infocom 2004, vol 1, 7–11 March 2004, Hong-Kong
70. Ott J, Kutscher D (2005) A disconnection-tolerant transport for drive-thru internet environments. In: Proceedings IEEE INFOCOM 2005. 24th Annual joint conference of the IEEE computer and communications societies, vol. 3, IEEE, pp 1849–1862
71. Papastergiou G, Psaras I, Tsaoussidis V (2009) Deep-space transport protocol: a novel transport scheme for space DTNs. Comput Commun Spec Issue Comput Communicationson Delay Disruption Tolerant Netw 32(16):1757–1767
72. Papastergiou G, Samaras C, Tsaoussidis V (2010) Where does transport layer fit into space dtn architecture? In: Advanced satellite multimedia systems conference (asma) and the 11th signal processing for space communications workshop (spsc), 2010 5th, IEEE, pp 81–88
73. Pentland A, Fletcher R, Hasson A (2004) Daknet: rethinking connectivity in developing nations. Computer 37(1):78–83

74. Pereira P, Casaca A, Rodrigues J, Soares V, Triay J, Cervelló-Pastor C (2011) From delay-tolerant networks to vehicular delay-tolerant networks. Commun Surv Tutorials IEEE 99:1–17
75. Postel J (1980) User datagram protocol. In: RFC-768, http://www.ietf.org/rfc/rfc768.txt (Last visited April 23, 2013)
76. Postel J (1981) Transmission control protocol. In: RFC-793, http://www.ietf.org/rfc/rfc793.txt (Last visited April 23, 2013)
77. Price K, Storn R (1997) Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces. J Global Optimiz 11:341–359
78. Psaras I, Papastergiou G, Tsaoussidis V, Peccia N (2008) DS-TP: Deep-space transport protocol. In: Aerospace conference, 2008 IEEE, pp 1–13
79. Psaras I, Wood L, Tafazolli R (2010) Delay-/disruption-tolerant networking: State of the art and future challenges. Technical Report, University of Surrey, UK
80. Ramadas M, Burleigh S, Farrell S (2008) Licklider transmission protocol—motivation. In: RFC-5325, http://tools.ietf.org/html/rfc5325 (Last visited April 23, 2013)
81. Ramadas M, Burleigh S, Farrell S (2008) Licklider transmission protocol specification. In: RFC-5326, http://tools.ietf.org/html/rfc5326 (Last visited April 23, 2013)
82. Samaras C, Tsaoussidis V (2008) DTTP: a delay-tolerant transport protocol for space internetworks. In: Proc. of second ERCIM Workshop on E-Mobility, pp. 3–14, May 30, Tampere, Finland
83. SSTL: Surrey Satellite Technology Ltd., http://www.sstl.co.uk
84. Sarkar M, Shukla KK, Dasgupta KS (2011) Article:a survey of transport protocols for deep space communication networks. Int J Comput Appl 31(8):25–32, published by Foundation of Computer Science. New York, USA
85. Schütz S, Eggert L, Schmid S, Brunner M (2005) Protocol enhancements for intermittently connected hosts. ACM SIGCOMM Comput Commun Rev 35(3):5–18
86. Scott K, Burleigh S (2007) Bundle protocol specification. In: RFC-5050
87. Seligman M, Fall K, Mundur P (2006) Alternative custodians for congestion control in delay tolerant networks. In: Proceedings of the 2006 SIGCOMM workshop on challenged networks. ACM , pp 229–236
88. Seligman M, Fall K, Mundur P (2007) Storage routing for dtn congestion control. Wirel Commun Mob Comput J, Wiley 7(10):1183–1196
89. Shokrollahi MA (2003) Raptor codes. In: IEEE international symposium on information theory
90. Small T, Haas Z (2003) 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, ACM, pp 233–244
91. Vahdat A, Becker D (2000) Epidemic routing for partially-connected ad hoc networks. In: Techical Report CS-200006, Duke University
92. Vahdat A, Becker D et al (2000) Epidemic routing for partially connected ad-hoc networks. Technical Report, Technical Report CS-200006, Duke University
93. Vellambi B, Subramanian R, Fekri F, Ammar M (2007) Reliable and efficient message delivery in delay tolerant networks using rateless codes. In: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking, ACM, pp 91–98
94. Wang R, Taleb T, Jamalipour A, Sun B (2009) Protocols for reliable data transport in space internet. Commun Surv Tutorials IEEE 11(2):21–32
95. Wang Y, Jain S, Martonosi M, Fall K (2005) Erasure-coding based routing for opportunistic networks. In: Proceedings of the 2005 ACM SIGCOMM workshop on delay-tolerant networking, ACM, pp 229–236
96. Wang Y, Wu H (2007) Delay/Fault-Tolerant mobile sensor network (DFT-MSN): a new paradigm for pervasive information gathering. IEEE Trans Mob Comput 6(9):1021–1034
97. Whitt W (2002) Stochastic-process limits. Springer, Heidelberg
98. Widmer J, Le Boudec J (2005) Network coding for efficient communication in extreme networks. In: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking, ACM , pp 284–291

99. Wood L, Eddy W, Holliday P (2009) A bundle of problems. In: Aerospace conference, 2009 IEEE, pp 1–17
100. Wood L, McKim J, Eddy W, Ivancic W, Jackson C (2009) Saratoga: a scalable file transfer protocol. Network Working Group Internet-Draft, http://tools.ietf.org/html/draft-wood-tsvwg-saratoga-10 (Last visited April 23, 2013)
101. Zhang Q, Jin Z, Zhang Z, Shu Y (2009) Network coding for applications in the delay tolerant network (dtn). In: 5th international conference on Mobile ad-hoc and sensor networks, 2009. MSN '09, pp 376–380
102. Zhang X, Neglia G, Kurose J, Towsley D (2006) On the benefits of random linear coding for unicast applications in disruption tolerant networks. In: 2006 4th international symposium on modeling and optimization in mobile, ad hoc and wireless networks, IEEE, pp 1–7
103. Zhang X, Neglia G, Kurose J, Towsley D (2007) Performance modeling of epidemic routing. Comput Netw 51:2867–2891

Chapter 11
Opportunistic Routing in Wireless Mesh Networks

Amir Darehshoorzadeh, Llorenç Cerdà-Alabern and Vicent Pla

Abstract Opportunistic Routing (OR) has been proposed as a way to increase the performance of wireless networks by exploiting its broadcast nature. In OR, instead of pre-selecting a single specific node to be the next-hop as a forwarder for a packet, multiple nodes can potentially be selected as the next-hop forwarder. Thus the source can use multiple potential paths to deliver the packets to the destination. More specially, when the current node transmits a packet, all the candidates that receive the packet successfully will coordinate with each other to determine which one would actually forward the packet according to some criteria, while the other nodes will simply discard the packet. In this chapter, we survey the state of the art in OR, then focus on the candidates selection algorithms and carry out a comparative performance evaluation of the most relevant proposals appeared in the literature.

Keywords Opportunistic routing · Candidate selection algorithms · Performance modeling

11.1 Introduction

In recent years, Multi-hop Wireless Networks (MWNs) have already become very popular and are receiving an increasing amount of attention by the research community. Compared to wired networks, routing in MWNs is specially challenging because

---

A. Darehshoorzadeh (✉) · L. Cerdà-Alabern
Computer Architecture Department, University Politècnica de Catalunya, Catalunya, Spain
e-mail: amir@ac.upc.edu

L. Cerdà-Alabern
e-mail: llorenc@ac.upc.edu

V. Pla
Communications Department, University Politècnica de València, València, Spain
e-mail: vpla@dcom.upv.es

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_11,

of two fundamental differences. The first one is the heterogeneous characteristics of the wireless links: due to the strong dependency of radio transmission impediments between the nodes with their distance and the environmental elements influencing the radio waves propagation. As a consequence, packet delivery probabilities may be significantly different for every link of a MWN. The second one is the broadcast nature of wireless transmissions: unlike wired networks, where links are typically point to point, when a node transmits a packet in a wireless network, this can simultaneously be received by several neighboring nodes.

Traditional routing protocols proposed for wireless networks perform best path routing, i.e., preselect one fixed route before transmissions starts. Each node in a route uses a fixed neighbor to forward to. This way, in the routing table of every node participating in the routing between a source and a destination, there is a forwarding entry which points to a neighbor (referred to as *next-hop*), over which packets addressed to the destination will be sent. Note that once all next-hops have been chosen, all packets between a source and destination follow the same path. This motivates the name of *uni-path* routing for such type of protocols. These approaches borrowed from the routing protocols for wire-line networks, and do not adapt well to the dynamic wireless environment where transmission failures occur frequently.

Opportunistic Routing, also referred to as diversity forwarding [25], cooperative forwarding [18] or any-path routing [16], is being investigated to increase the performance of MWNs by taking advantage of its broadcast nature. In OR, in contrast to traditional routing, instead of preselecting a single specific node to be the next-hop as a forwarder for a packet, an ordered set of nodes (referred to as *candidates*) is selected as the potential next-hop forwarders. We shall refer to the ordered set of candidate of a node as its Canditates Set (CS). Thus, the source can use multiple potential paths to deliver the packets to the destination. More specifically, when the current node transmits a packet, all the candidates that successfully receive it will coordinate with each other to determine which one will actually forward it, while the others will simply discard the packet.

For a better understanding of the inherent benefits associated to Opportunistic Routing, consider the example shown in Fig. 11.1 (the example has been taken from [5]). It presents the possibility that one transmission may reach a node which is closer to the destination than the particular next-hop in traditional routing.

Assume that S is the source and D is the destination and the packet transmissions in each link are Bernoulli with the delivery probabilities a specified over the links. If

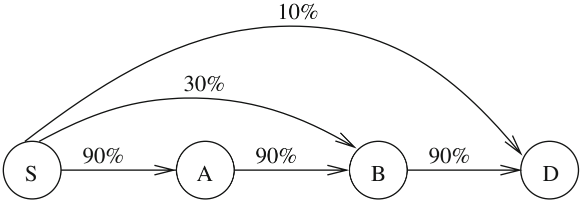


Fig. 11.1 An example of opportunistic routing

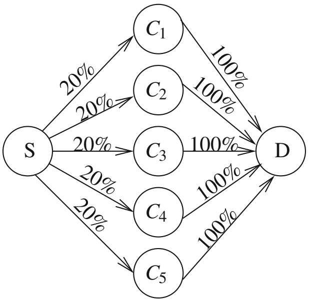


Fig. 11.2 An illustration of virtual link in opportunistic routing

a packet sent by S is correctly received by B but node A, it has to be retransmitted by S until it reaches the designated next-hop A. Another situation that might happen is that a packet sent by S is correctly received by both nodes A and B. Although node B is closer to the destination than node A, it is not allowed to forward the packet. In contrast to the traditional routing, OR takes advantage of any these situations to maximize the packet progress toward the destination. An OR protocol can use $\{D, B, A\}$ as the candidates (D is the highest priority candidate, and A the least one) to forward the packet. If both nodes A and B receive the packet but not D, since node B has more priority than A (it is closer to the destination), then it will forward the packet while node A will simply discard it.

Another benefit of Opportunistic Routing is that it increases the reliability of transmissions by combining weak physical links into one strong *virtual link*. In other words, it acts like OR has additional backup links and the possibility of transmission failure is reduced [19]. As shown in Fig. 11.2, the sender has a low delivery probability to all its neighbors, while they have a perfect link to the destination. Under a traditional routing protocol, we have to pick one of the five intermediate nodes as the relay node. Thus, altogether we need 5 transmissions on average to send a packet from the source to the relay node and 1 transmissions from the relay node to the destination. In comparison, under OR, we can select the five intermediate nodes as the candidates. The combined link has a success rate of $(1 - (1 - 0.2)^5) \approx 0.67$. Therefore, on average only $1/0.67 = 1.48$ transmissions are required to deliver a packet to at least one of the five candidates, and another transmission is required for a candidate to forward the packet to the destination, so on average it takes only 2.48 transmissions to deliver a packet to the destination.

11.2 Issues in Opportunistic Routing

Three main issues arise in the design of OR protocols:

- **Candidate selection**. All nodes in the network must run an algorithm for selecting and sorting the set of neighboring nodes (candidates) that can better help in the forwarding process to a given destination. We shall refer to this algorithm as

candidate selection. The aim of candidate selection algorithms is to minimize the expected number of transmissions from the source to the destination. In Sect. 11.5 some noteworthy candidate selection algorithms are described.
- **Opportunistic Routing metric**. In order to accurately select and prioritize the CSs, Opportunistic Routing algorithms require a metric. First Opportunistic Routing algorithms were based on simple metrics inherited from traditional unicast routing, as those used by Shortest Path First (SPF) algorithms. However, some researchers realized that more accurate metrics were required in OR. Different metrics in Opportunistic Routing are discussed in detail in Sect. 11.3.
- **Candidate coordination**. It is the mechanism used by the candidates to discover which one has the highest priority that has received, and thus, must forward, the packet. Coordination requires signaling among the nodes, and imperfect coordination may cause duplicate transmission of packets. Different approaches of candidate coordination are presented in Sect. 11.4.

11.2.1 Research Directions in Opportunistic Routing

In this section we give an overview of the main research contributions in Opportunistic Routing. Table 11.1 shows some of the OR research contributions found in the literature that will be described in the following sections. The meaning of the columns is the following:

- **Protocol**: Here there is the name of the protocol coined by the authors, or NA if no name was given. The corresponding reference is also provided here.
- **Year**: Year of publication.
- **Type**: Method to obtain the numerical results presented in the paper. We use the keys: *S*, for simulation; *A*, for analytical; and *E*, for Experimental.
- **Topic**: Main topic of the paper.
- **Metric**: Metric used by the candidate selection algorithm (see Sect. 11.3).
- **Coord.**: Coordination method used in the paper (see Sect. 11.4). The table shows *NA* in those papers where a perfect coordination is assumed without relying in any specific type of coordination.
- **Cand. Sel.**: Information used by the candidate selection algorithm (see Sect. 11.5): *Topology* when it is related with the topological graph of the network, and *Location* when it uses the geographical position of the nodes.

Entries in Table 11.1 are sorted in chronological order. The table shows the increasing interest that has emerged related with OR in the last decade.

Most of the research in Opportunistic Routing is related with the issues described in Sect. 11.2, but there are other areas of OR that have been investigated as well. We have identified the following as the main topics on research in Opportunistic Routing:

- Metrics, Sect. 11.3.
- Candidate coordination, Sect. 11.4.

Table 11.1 Classification of research works in opportunistic routing protocol

Protocol	Year	Type	Topic	Metric	Coord.	Cand. Sel.
SDF [26]	2001	S	Candidate coordination	ETX	Ack	Topology
GeRaF [53]	2003	A/S	Candidate coordination	Geo.	RTS-CTS	Location
ExOR ver-1 [4]	2004	S	Candidate selection	ETX	Ack	Topology
ExOR ver-2 [5]	2005	E	Candidate coordination	ETX	Timer	Topology
NA [38]	2005	A/S	Sensor networks	Geo.	RTS-CTS	Location
COPE [21, 22]	2005	E	Network coding	ETX	Net. coding	Topology
OAPF [52]	2006	S	Candidate selection	ETX/EAX	Ack	Topology
LCOR [15, 17]	2007	S	Candidate selection	EAX	NA	Topology
MORE [10]	2007	E	Network coding	ETX	Net. coding	Topology
GOR [50]	2007	S	Candidate selection	Geo.	Timer	Location
NA [35]	2008	A	Analytical	Geo.	NA	Location
NA [3]	2008	A/S	Analytical	Geo.	NA	Location
NA [18]	2008	E	Candidate selection	ETX	Ack	Topology
CORE [45, 46]	2008	S	Network coding	Geo.	Timer	Location
MTS [31]	2009	S	Candidate selection	EAX	Timer	Topology
POR [47]	2009	S	Candidate selection	Geo.	Timer	Location
SOAR [37]	2009	S/E	Candidate selection	ETX	Timer	Topology
Pacifier [23]	2009	S	Multicast	ETX	Net. coding	Topology
NA [8]	2010	A	Analytical	EAX	NA	Location
MSTOR [30]	2010	S	Multicast	EAX/ETX	Ack	Topology
MORP [11]	2011	S	Multicast	ETX	Ack	Topology
NA [13]	2011	A	Analytical/cand. selec.	ETX/EAX	NA	Topology

NA in the column **Coord.**: Perfect coordination

- Candidate selection algorithms, Sect. 11.5.
- Network Coding (NC), Sect. 11.6.
- Geographic Opportunistic Routing, Sect. 11.7.
- Multicast Opportunistic Routing, Sect. 11.8.
- Sensor networks.

Although many of the OR proposals can be adapted for sensors networks, there are some contributions that specifically study Opportunistic Routing in this context. As an example, we have included [38] in Table 11.1. In this paper the authors take into consideration how OR can be exploited when there are the characteristic power down periods that occur in sensor networks. Due to the limited number of works in this specific area, we do not analyze this topic further. The rest of topics will be addressed in the next sections as indicated above.

11.2.2 Motivation

In a wireless network, when a packet is unicast to a specific next-hop node of the sender at the network layer, all the neighboring nodes of the sender may be able to overhear the packet at the physical layer. It is possible that some of the neighbors may have received the packet correctly while the designated next-hop node did not. Opportunistic Routing has been proposed as a way to take the advantages of the spacial diversity and broadcast nature of wireless communications by selecting multiple nodes as the candidates for forwarding the traffic.

As mentioned in Sect. 11.2, the performance of Opportunistic Routing depends on different key issues. One of the important issues of Opportunistic Routing is candidate selection and relay priority assignment. All nodes in the network must run an algorithm for selecting and sorting the set of neighboring nodes (candidates) that can better help in the forwarding process to a given destination. Candidate coordination is another key issues in Opportunistic Routing, i.e. the mechanism used by the candidates to discover which is the highest priority candidate that has received, and thus, must forward, the packet.

This chapter carries out a comprehensive study on the state of the art in OR with emphasis on the candidates selection algorithms. In [19] authors surveyed Opportunistic Routing issues with emphasis on the candidate coordination algorithms. We also provide a Discrete Time Markov Chain(DTMC) to assess the improvement that may be achieved using Opportunistic Routing. We use the proposed model to carry out a comparative performance evaluation of the most relevant candidate selection algorithms in the literature.

After the sections explaining the work in the research areas listed in Sect. 11.2.1, we continue the chapter by describing analytical models of OR in Sect. 11.9. In Sect. 11.10 we presented some numerical results that illustrate the performance achieved with some relevant candidate selection algorithms described in Sect. 11.5. Finally, some concluding remarks and some suggestions for future research directions are given in Sect. 11.11.

11.3 Routing Metrics

The general aim of OR is to minimize the expected number of transmission required to carry a packet from the source to the destination. If the number of transmissions is reduced, the end-to-end delay, its variations, and the energy consumed will be reduced. The set of candidates which each node uses and their priority order have a significant impact on the performance that OR can achieve. Therefore, using a good metric to select and order the candidates is a key factor in designing an OR protocol.

Candidates in Opportunistic Routing can be prioritized based on hop count [20, 42, 49], geographic-distance [51, 53] (Geo-Distance), Expected Transmission Count (ETX) [14], Expected Any-Path Transmission (EAX) [52] and so on. Utilization of hop count, ETX or EAX needs an underlying routing protocol (either reactive or proactive) to gather such information. Geo-Distance requires the availability of location information of nodes. The accuracy of a metric depends on the proper measurement of link quality and timely dissemination of such information [33, 39]. Below, we describe the two usual metrics ETX and EAX that have been widely used in the literature.

Expected Transmission Count (ETX) [14]: is the average number of transmissions required to reliably send a packet across a link or route including retransmissions. The ETX of a single path route is the sum of the ETX for each link in the route. With the assumption of the packet transmission between nodes i and j as Bernoulli trials with delivery probability p_{ij}, the Expected Transmission Count of the link is:

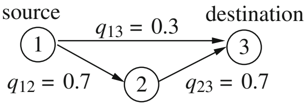
$$ETX(i, j) = \frac{1}{p_{ij}} \tag{11.1}$$

In Opportunistic Routing, however, it is necessary to consider the fact that there are some candidates which can receive the packet, thus, a packet may travel along any of the potential paths. Authors in [15, 31] have shown that using ETX may give suboptimal selection of candidates and in [34] it was shown that OR in combination with ETX could degrade the performance of the network. Because of that Zhong et al. [52] proposed another metric that has been widely adopted in Opportunistic Routing.

Expected Any-Path Transmission (EAX) [52]: is an extension of ETX and can capture the expected number of transmissions taking into account the multiple paths that can be used under OR. Several methods to compute EAX have been proposed by other authors [9, 15, 31]. In Sect. 11.10.1 we present a model for OR that can be used to calculate the expected number of transmissions from source to the destination.

The following simple example illustrates the meaning of Expected Any-Path Transmission (EAX). Consider the network topology and the Canditates Set (CS) of each node in Fig. 11.3. Node 1 is the source and node 3 is the destination. Assume that packet transmissions in each link are Bernoulli with the delivery probabilities from node i to node j, q_{ij}, indicated in the figure. Note that in the Canditates Set (CS) of node 1, node 3 has higher priority than node 2. Note also that node 2 has only one candidate (the destination). Therefore, upon being the next forwarder, node 2 would behave as in traditional routing. We now compute the expected number of transmissions from node 1 to the destination using Opportunistic Routing. We can write: $E_1 = 1 + \sum_{i=1}^{3} p_i E_i$, where p_i is the probability of node i being the next forwarder (or the destination), and E_i is the expected number of transmissions from node i to the destination (note that $E_3 = 0$). Grouping terms we have $E_1 = (1 + p_2 \times E_2)/(p_2 + p_3) = (1 + (1 - q_{13})\, q_{12} \times 1/q_{23})/((1 - q_{13})\, q_{12} + q_{13})$. Substituting we get $E_1 \approx 2.15$. Using ETX, the shortest path would yield $E_1^{ETX} = 2/0.7 = 2.86$. Thus, although Expected Transmission Count (ETX) is much simple to compute than Expected Any-Path Transmission (EAX), it does not accurately compute the expected number of transmissions under Opportunistic Routing (OR).

Fig. 11.3 Illustration of the EAX metric



Network topology.

11.4 Candidate Coordination Methods

One of the important issues of OR is the candidate coordination, i.e., the mechanism used by the candidates to discover which is the highest priority candidate that has received, and thus, must forward, the packet. It requires signaling between the nodes, and imperfect coordination may cause duplicate transmission of packets. A good coordination approach should select the best candidate without duplicate transmissions while using the smallest time or control overhead.

With perfect coordination among candidates, the larger is the number of candidates the lower is the expected number of transmissions from the source to the destination. However, increasing the number of candidates increases also the coordination overhead. Therefore, in practice, the maximum number of candidates that can be used is limited. This fact has often been neglected in candidate selection algorithms proposed in the literature. Perfect coordination and no signaling overhead has been assumed and the algorithms have been designed to select all possible candidates to reduce the expected number of transmissions.

Existing coordination approaches are divided into three main categories based on the mechanism used: acknowledgment-based (ACK-based), timer-based, Network Coding (NC) and Request-To-Send (RTS)-Clear-To-Send (CTS) Coordination. In the following sections we briefly describe these approaches.

11.4.1 Acknowledgment-Based Coordination

It is one of the first methods that was proposed for candidate coordination. Upon receiving a data packet, candidates send back a short acknowledgment (ACK) in decreasing order of candidate priority.

This method was first proposed in [25] as the coordination mechanism for the Selection Diversity Forwarding (SDF) protocol. In SDF coordination is achieved by means of a four-way-handshaking: the candidates receiving the data packet send back an acknowledgment to the sender. Based on the acknowledgments, the sender sends a forwarding order to the best candidate, which is also acknowledged.

A similar approach is used in Extremely Opportunistic Routing (ExOR) [4], which uses a modified version of the 802.11 MAC. It reserves multiple slots of time for the receiving nodes to return acknowledgments. Instead of only indicating that the packet was successfully received, each ACK contains the ID of the highest priority successful recipient known to the ACKs sender. All the candidates listen to all ACK slots before deciding whether to forward, in case a low-priority candidates ACK reports a high-priority candidate ID and whose ACK was not correctly received. Including the ID of the sender of the highest-priority ACK heard so far helps suppress duplicate forwarding. This strategy requires that candidates be neighbors of each other such that the transmission of an ACK can be overheard by all of them.

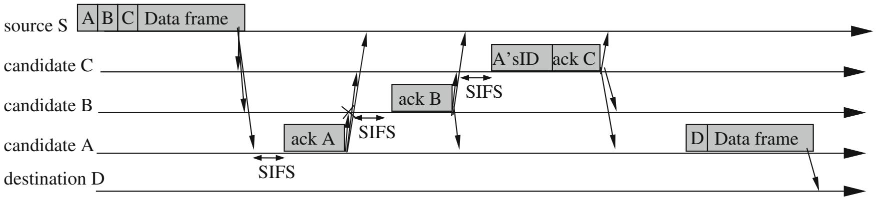


Fig. 11.4 Acknowledgment-based coordination using a modified 802.11 MAC

As an example of the ACK-based coordination, consider a network with source S and destination D. Assume that the CS of S is $\{A, B, C\}$ (A has the highest priority and C has the lowest). Suppose that all candidates receive a transmission from source. Figure 11.4 shows ACK-based coordination method for this example. All candidates transmit acknowledgments in decreasing order of candidate priority: the first acknowledgment slot belongs to node A, the second slot belongs to node B and the third slot is dedicated to C. In Fig. 11.4 we suppose that the acknowledgment from A does not receive by B, but node C does hear the A's ACK (see Fig. 11.4). Suppose further that node B hears node C's ACK. If ACKs did not contain IDs, node B would forward the packet, since to its knowledge it is the highest priority recipient. The fact that node C's ACK contains node A's ID indirectly notifies B that node A did receive the packet. Once node A has successfully determined itself as the responsible node, it forwards the packet.

11.4.2 Timer-Based Coordination

In this method, all candidate which are included in the packet are ordered based on a metric. After a data packet is broadcasted, candidate will respond in order, i.e., ith candidate will respond at the ith time slot. A candidate forwards data packet at its turn only when it does not hear other candidates that forward the packet. Thus, when a candidate forwards a data packet, it means that all other higher priority candidates failed to receive the data packet. In another word, forwarding a data packet by a candidate will prevent the lower priority ones to forward it. In the example of Fig. 11.1, assume that $\{B, A\}$ is the CS of source S to reach destination D (B is the better candidate and given the higher priority). After receiving the packet sent by S, candidate B forwards the packet in the first time slot, while A schedules to transmit in the second time slot. If A is in the range of B, overhearing the data packet sent from B by A means that a higher priority candidate received the packet and has forwarded it, thus A simply discard the packet.

This approach is simple and easy to implement and no control packet is required. The overhead of the timer-based coordination is candidate waiting time. The main drawback of this solution is duplicate transmission because of not all candidates are guaranteed to overhear the forwarding from the selected candidate [19].

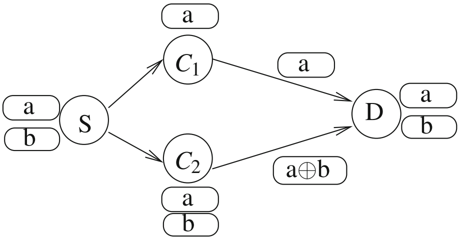


Fig. 11.5 NC coordination approach [33]

11.4.3 Network Coding Coordination

Another approaches to prevent duplicate transmission is combining OR with NC [2] which provides an elegant method for candidate coordination [6, 10, 45]. The basic principle behind combining NC with Opportunistic Routing is that forwarders can combine the packets to be transmitted so as to deliver multiple data packets through a single transmission. When transmitting packets from source to a destination, a flow is divided into batches which contain several native packets (original packets without coding). The source broadcasts random linear combinations of native packets, and candidates forward the linear combinations of received coded packets. When the destination has enough linearly independent coded packet, then it can decode them to reconstruct the set of initial packets.

In order to better clarify the advantage of combining NC with Opportunistic Routing, consider the example in Fig. 11.5. Assume that source S transmits two packets a and b using CS $\{C_1, C_2\}$. Assume that C_2 receives both packets but C_1 receives only one of them (see Fig. 11.5). Node C_1 transmits first because it is closer than C_2 to the destination. Node C_2 has the following three choices: forwarding a, b, or both a and b. In the NC, node C_2 can forward a coded packet $a \oplus b$. When D receives transmitted packets from C_1 and C_2, it can decode and restore the original packets. It performs an XOR operation on the two received packets: $a \oplus b \oplus b = a$. Thus, no duplicate transmission occurs at D.

However, using NC with OR may lead to a high number of potential forwarders sending coded packets, and thus, resulting in redundant transmissions. There exists a trade-off between transmitting a sufficient number of coded packets to guarantee that the destination has enough coded packets to reconstruct the native packets, and avoiding to inject the network with unnecessary packets [6].

11.4.4 RTS-CTS Coordination

Some other mechanisms like [20, 53] use explicit control packet(s) exchanged immediately before sending a data packet. In this approach the sender multicasts the RTS to the its CS (it is actually a broadcast control packet). The RTS contains all the

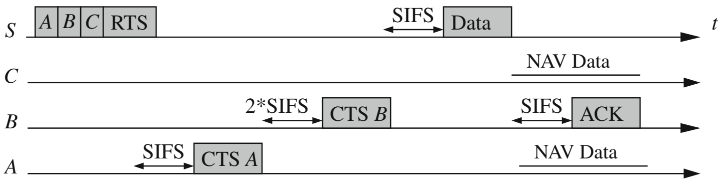


Fig. 11.6 RTS-CTS coordination approach

candidates addresses which are ordered according to a metric. When an intended candidate receives the RTS packet, it responds by a CTS. These CTS transmissions are sent in decreasing order of candidate priority: the first candidate in priority transmits the CTS after a Short Interframe Space (SIFS), the second one after 2× SIFS, and so on. When the sender receives a CTS, it transmits the DATA packet to the sender of this CTS (which would be the highest priority candidate that responded) after a SIFS interval. This ensures that other lower priority candidates hear the DATA before they send CTS and suppress any further CTS transmission. All such receivers then set their Network Allocation Vector (NAV) until the end of ACK period. This mechanism is guaranteed to have a single winner and it can avoid duplicate transmissions.

Figure 11.6 shows an example of RTS-CTS coordination. Assume that there are three candidates A, B and C to reach the destination (A the highest priority candidate and C the least one). After receiving RTS by candidates they send the CTS packet in order of their priorities. Here we assume that the first CTS which belong to A was not received, but the second one was received. When the sender S receives the first CTS from B, it sends the data packet to it, therefore the highest priority candidate whose its CTS is received by the source will forward the data packet.

11.5 Candidate Selection Algorithms

Another important component of OR is candidate selection, which is similar to building routing tables in traditional routing. Selection of good candidates can affect the performance of the network. According to the amount of information that is needed to select and prioritize the candidates, candidate selection algorithms can be divided into two categories: Location-based and Topology-based selection. In location-based selection [53], each node maintains a limited state information and independently determines its own CS along the path to the intended destination. Topology-based selections [4, 5] find the CSs according to the global topology information of the network. Therefore, a node requires to maintain network state information, for example, the network topology, state information on each link, and flow-related information (e.g., path and data rate), what can run into a scalability problem. In general, topology-based strategy outperforms location-based strategy, since the former can optimize the selection of a CSs with more network state information gathered. However, the location-based strategy might be easier to implement, requires less signaling and scales better than topology-based [33]. The authors in [28] proposed a new routing paradigm, *plasma*, for wireless multihop networks. In plasma routing the choice of the path and gateway for each packet is not made beforehand by the source node, but rather on-the-fly by the mesh routers as the packet traverses the network. In their work, a distributed routing algorithm to jointly optimize the transmission rate and the set of gateways are proposed. They also proposed a load balancing technique to disperse the network traffic among multiple gateways.

In this section, we describe four different candidate selection algorithms that have been proposed in the literature. They range from non-optimum, but simple, to optimum, but with a high computational cost. These algorithms are: Extremely Opportunistic Routing (ExOR) [4]; Opportunistic Any-Path Forwarding (OAPF) [52]; Least-Cost Opportunistic Routing (LCOR) [15, 17]; and Minimum Transmission Selection (MTS) [31].

ExOR is one of the firsts and most referenced Opportunistic Routing protocols, it is based on ETX and is simple to implement. OAPF has an intermediate complexity: it uses the EAX metric but it does not guarantee to yield the optimal sets of candidates (i.e. the CSs that minimize the expected number of transmissions). Finally, we have chosen LCOR and MTS which select the optimal sets of candidates.

Here we introduce some notations that we use throughout this section:

- $ncand$ is the maximum number of candidates per node.
- $ETX(v, d)$ is the uni-path ETX between the two nodes v and d.
- $EAX(C_{v,d}, v, d)$ is the EAX between node v and d by using $C_{v,d}$ as the CS of v to reach node d.
- $N(v)$ is the set of all neighbors of node v.
- $|S|$ is the cardinality of the set S.

In the following sections we describe each algorithm with more details. For the sake of being precise, we shall give a pseudo-code for each algorithm.

To show the differences between ExOR, OAPF, LCOR and MTS, we use a simple example shown in Fig. 11.7. In this example node S is the source; D is the destination; the number on each link indicates its packet delivery probability (symmetric) and the maximum number of candidates for each node is set to 2 ($ncand$=2). The ETX using uni-path routing from each node to the destination D is shown in Table 11.2.

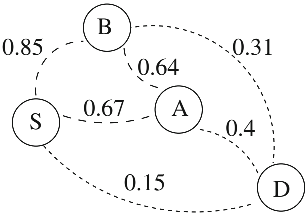


Fig. 11.7 An example of Candidate selection

11.5.1 Extremely Opportunistic Routing

Biswas and Morris proposed ExOR [4], one of the firsts and most referenced OR protocols. The selection of candidates is based on the ETX metric. The basic idea of ExOR is running the Shortest Path First (SPF) with $weight = 1/q_{ij}$, where q_{ij} is the delivery probability of the link between the two nodes i and j (i.e. the weights are the ETXs of the links).

The algorithm for selection of candidates in ExOR is shown in Algorithm 1. Every node s except the destination d runs this algorithm. The first node after s in the shortest path is selected as candidate ($cand$) if its ETX to the destination (ETX($cand, d$)) is less than ETX(s, d). Then the link between s and $cand$ is removed, and this process is repeated until no more paths to d are available, or the maximum number of candidates is reached ($|C_{s,d}| = ncand$). Finally, ETX($cand, d$) (or 0 if the $cand$ is the destination) is used to sort the CS.

Assume that node S in Fig. 11.7 want to find its CS using ExOR. According to ExOR's algorithm (see Algorithm 1), node S finds the SPF to D which is S-A-D with ETX = 3.99 (see Table 11.2). Therefore, node A is the first candidate for node S. Then, edge S-A is removed from the topology and SPF is run again. The new shortest path from S to D is S-B-D with ETX = 4.40 and B is selected as the next candidate. Finally, the ETX of each candidate to the destination d is used to sort the CS. The final candidates set of node S is $C_{S,D} = \{A, B\}$. ExOR uses ETX to estimate the closeness to the destination but, this metric does not account for the fact that packets are delivered by the candidates under opportunistic forwarding.

There is a second version of ExOR [5] proposed in 2005. To cope with the acknowledgment and the coordination issue, ExOR adopts batch transmission; 10–100

Algorithm 1: Candidate.selection.ExOR(s, d, $ncand$).

```
$G_{tmp}$ ← temporal copy of the network topology
$cost(s) \leftarrow ETX(s, d)$ in $G_{tmp}$
$C_{s,d} \leftarrow \emptyset$
while $|C_{s,d}| < ncand$ & $(s, d)$ connected in $G_{tmp}$ do
    $cand \leftarrow$ first node after $s$ in the $SPF(s, d)$ in $G_{tmp}$
    if $cand == d$ then
        $C_{s,d} \leftarrow C_{s,d} \cup \{d\}$
        $cost(cand) \leftarrow 0$
    else
        $cost(cand) \leftarrow ETX(cand, d)$ in $G_{tmp}$
        if $cost(cand) < cost(s)$ then
            $C_{s,d} \leftarrow C_{s,d} \cup \{cand\}$
        end
    end
    $G_{tmp} \leftarrow$ delete $edge(s, cand)$ in $G_{tmp}$
end
$C_{s,d} \leftarrow C_{s,d}$ ordered by $cost$
```

Table 11.2 Expected number of transmissions of each node to D in Fig. 11.7

Node	ETX(Node,D)
S	3.99
A	2.5
B	3.22
D	0

packets are collected in a batch to transfer and the next batch starts only when the current batch has completed. For each packet of the same batch, the source node selects a CS, which are prioritized by closeness to the destination. The closeness property of a node is evaluated employing the ETX metric.

Each packet has a batch map. For each packet in the batch, this map indicates the highest-priority node known to have received a copy of that packet. Then, as the packet progresses towards the destination the batch map contained in the packet is used to update the local batch maps stored in the receiving nodes. A forwarder is allowed to broadcast only received packets that its local batch map indicates have not been forwarded by any other higher priority node. Therefore, the coordination is done using timer-based coordination (see Sect. 11.4.2). The evaluation was performed on Roofnet [1], an outdoor roof-top 802.11b network. This version of ExOR guarantees to transmit 90 % of a batch using opportunistic forwarding, while the remaining packets are sent with traditional unicast routing.

Simple Opportunistic Adaptive Routing (SOAR) [37] has been proposed after ExOR. In order to leverage path diversity while avoiding duplicate transmissions, SOAR relaxes the actual route that data traverses to be along or near the default path but constrains the nodes involved in routing a packet to be near the default path. Moreover, this forwarding node selection also simplifies coordination since all the nodes involved are close to nodes on the default path and can hear each other with a reasonably high probability. It selects the shortest path between source and destination using ETX, and the nodes near to the shortest path can act as the CS. SOAR uses timer-based approach for candidate coordination.

11.5.2 Opportunistic Any-Path Forwarding

OAPF [52] is an Opportunistic Routing protocol which is based on ETX and EAX. The pseudo-code of OAPF is shown in Algorithm 2.

In OAPF the selection of candidates can be performed as follows at a node s for a specific destination d. First, a set of initial candidates $\hat{C}_{s,d}$ is determined based on the best path ETX. A neighbor v is included in the initial CS ($\hat{C}_{s,d}$) only if ETX(v, d) $<$ ETX(s, d). Then, a subset of $\hat{C}_{s,d}$ is selected as the actual Canditates Set, $C_{s,d}$. Note that, all nodes in the initial CS must select their CSs before s.

After initiating the CS, s selects the best candidate among the nodes in its initial Canditates Set. Here, the best candidate is the one that reduces the most the expected

Algorithm 2: Candidate.selection.OAPF(s, d, $ncand$).

```
1  $C_{s,d} \leftarrow \emptyset$
2  $\hat{C}_{s,d} \leftarrow \emptyset; m_p \leftarrow \infty$
3  if $s == d$ then
4      $cost(s) \leftarrow 0$
5      return
6  end
7  forall the $v \in N(s)$ do
8      if $ETX(v,d) < ETX(s,d)$ then
9          $\hat{C}_{s,d} \leftarrow \hat{C}_{s,d} \cup \{v\}$
10     end
11 end
12 while $|C_{s,d}| < ncand$ // search for the best candidate
13 do
14     $cand \leftarrow \arg\min_{c \in \hat{C}_{s,d}} EAX(C_{s,d} \cup \{c\}, s, d)$
15     $m_c \leftarrow EAX(C_{s,d} \cup \{cand\}, s, d)$
16     if $m_c < m_p$ then
17         $C_{s,d} \leftarrow C_{s,d} \cup \{cand\}$
18         $\hat{C}_{s,d} \leftarrow \hat{C}_{s,d} \setminus \{cand\}$
19         $m_p \leftarrow m_c$
20     else
21         $cost(s) \leftarrow m_p$
22         break
23     end
24 end
25 $C_{s,d} \leftarrow C_{s,d}$ ordered by $cost$
```

number of transmissions from s to the destination (line 14). Node s adds the best candidate to its actual Canditates Set ($C_{s,d}$) and removes it from its initial set. Note that to find the best candidate in each iteration, candidates should be ordered according to their EAX. It tries again to find the best node from its new initial Canditates Set. This process is repeated until there is not any other suitable node to be included in the CS of s, or the number of candidates in the $C_{s,d}$ reaches the maximum number of candidates ($ncand$). Finally, the CS is ordered by EAX of each candidate.

Now assume that node S in Fig. 11.7 wants to find its CS using OAPF. First, it creates its initial Canditates Set $\hat{C}_{S,D}$. Since the ETX of all its neighbors (A, B and D) to the destination D is less than ETX(S,D) (see Table 11.2) then, the initial Canditates Set of S is $\hat{C}_{S,D} = \{A, B, D\}$. Note that, all nodes in the initial Canditates Set must select their CSs before S. In Table 11.3 we summarize the Canditates Set and related expected number of transmissions for node A and B.

Table 11.3 Candidates set of A and B in Fig. 11.7 using OAPF

Node	Candidates set	EAX
A	$\{D\}$	2.5
B	$\{D, A\}$	2.79

Table 11.4 OAPF Operation

Iteration	Selection		
1	$EAX(\{A\}, S, D)$=3.99,	$EAX(\{B\}, S, D) = \mathbf{3.97}$,	$EAX(\{D\}, S, D)$=6.66
2	$EAX(\{A, B\}, S, D)$= 3.64,	$EAX(\{D, B\}, S, D) = \mathbf{3.46}$	

Table 11.4 shows the process of selecting candidates for the source S using OAPF. In the first iteration source selects B as its candidate. Because B is the one that reduces the expected number of transmissions from S to D the most. Then, node B is removed from initial Canditates Set. The CS of S in the first iteration would be $C_{S,D} = \{B\}$. In the section iteration of while-loop in Algorithm 2 (line 12–24), source looks for the second candidate from the remaining potential candidates in $\hat{C}_{S,D} = \{A, D\}$. As we can see in Table 11.4, the second candidates that reduces the expected number of transmissions from S to D the most is D. Therefore the final CS for source using OAPF is $C_{S,D} = \{D, B\}$ with EAX equal to 3.46.

11.5.3 Least-Cost Opportunistic Routing

The goal of this algorithm is to find the optimal CSs. Recall that the optimal CSs are the sets that minimize the expected number of transmissions from the source to the destination. LCOR [15] uses EAX as the metric to select candidates as shown in Algorithm 3. It works similar to the classical distributed Bellman-Ford algorithm.

Algorithm 3: Candidate.selection.LCOR(s, d, $ncand$).

```
forall the $v$ in the network $\setminus\{d\}$ do
    $cost_{curr}(v) \leftarrow \infty$; $cost_{prev}(v) \leftarrow \infty$
end
$cost_{curr}(d) \leftarrow 0$
repeat
    flag ← TRUE
    forall the $v$ in the network $\setminus\{d\}$// search for the best CS
    do
        $C_{v,d} \leftarrow \underset{S \in 2^{|N(v)|}, 0<|S|\le ncand}{\arg\min} EAX(S, v, d)$
        $cost_{curr}(v) \leftarrow EAX(C_{v,d}, v, d)$
    end
    forall the $v$ in the network $\setminus\{d\}$ do
        if $cost_{curr}(v) \neq cost_{prev}(v)$ then
            $cost_{prev}(v) \leftarrow cost_{curr}(v)$
            flag ← FALSE
        end
    end
until $flag == TRUE$
$C_{s,d} \leftarrow C_{s,d}$ ordered by $cost_{curr}$
```

The algorithm proceeds iteratively and at each iteration an exhaustive search over all possible CSs is carried out. It starts by initializing the cost (EAX) of each node v to reach the destination d (lines 1–4). Since in the initializing phase the CSs for all nodes are empty, the cost to reach the destination for all nodes is equal to ∞ ($cost_{curr}(v) \leftarrow \infty$). Note that the cost for the destination d is always equal to 0 ($cost_{curr}(d) \leftarrow 0$).

To find the optimal Canditates Sets in each iteration, and for every node v except the destination, the algorithm runs an exhaustive search over all possible subsets of $N(v)$ with cardinality not exceeding $ncand$ (line 9). The algorithm terminates when the cost to reach the destination does not change for all nodes in two consecutive iterations (lines 12–17).

In each iteration the algorithm checks for all the nodes but the destination, all subsets of their neighbors with cardinality $\leq ncand$. Therefore, for dense networks the computational cost of the algorithm increases extremely fast due to the combinatorial explosion of the exhaustive search of line 9. In Sect. 11.10.5.4, we will carry out an experimental evaluation of the computational time.

Applying LCOR on the topology in Fig. 11.7 yields as a result $C_{S,D} = \{D, A\}$ with the expected number of transmissions equal to 3.36.

11.5.4 Minimum Transmission Selection

MTS [31, 48] is another algorithm which selects the optimal CSs for any node to a given destination d that minimizes the total expected number of transmissions. Like LCOR, this algorithm proceeds iteratively and uses EAX as the metric for selecting the CSs.

The general idea of MTS consists of moving from the nodes closest to the destination d (in terms of the EAX) backwards to the source. Note that, the closest node to the destination has the least EAX. MTS uses the following principle: if u and v are neighbors and $EAX(C_{u,d}, u, d) < EAX(C_{v,d}, v, d)$, then adding u and its candidates to the CS of node v will reduce the expected number of transmissions from v to d, i.e. $EAX(C_{v,d} \cup \{u\} \cup C_{u,d}, v, d) < EAX(C_{v,d}, v, d)$.

Given a general wireless topology, for a given destination d initially let $\mathbb{S}$ be the set of all nodes except d. The MTS algorithm for computing the optimal CS from any source node $v \in \mathbb{S}$ to d is described in pseudo-code in Algorithm 4. The algorithm starts by initializing the cost (EAX) of each node v to reach the destination d (lines 2–10 in Algorithm 4). If d is one of the neighbors of v, then v adds the destination to its CS and the cost to reach the destination ($cost(v)$) is set to $\frac{1}{q_{vd}}$, where q_{vd} is the delivery probability of link between the two nodes v and d (note that $EAX(C_{v,d}, v, d) = \frac{1}{q_{vd}}$ when $C_{v,d} = \{d\}$).

At each subsequent iteration while $\mathbb{S}$ is not empty the algorithm looks for the node $minnode$ with minimum cost in terms of the expected number of transmissions to the destination (line 11). The neighbors of $minnode$, $N(minnode)$, add $minnode$ and

Algorithm 4: Candidate.selection.MTS($\mathbb{S}$, d,$ncand$).

```
   Data: $\mathbb{S}$ is the set of all nodes except $d$.
1  $cost(d) \leftarrow 0$
2  forall the $v \in \mathbb{S}$ do
3      if $v \in N(d)$ then
4          $cost(v) \leftarrow \frac{1}{q_{vd}}$
5          $C_{v,d} \leftarrow d$
6      else
7          $C_{v,d} \leftarrow \emptyset$
8          $cost(v) \leftarrow \infty$
9      end
10 end
11 while $\mathbb{S}$ is not empty do
12     $minnode \leftarrow \arg\min_v cost(v)$
13     $\mathbb{S} \leftarrow \mathbb{S} \setminus \{minnode\}$
14     forall the $v \in N(minnode)$ do
15         $C_{v,d} \leftarrow merge(C_{v,d}, minnode, C_{minnode,d})$
16         $cost(v) \leftarrow EAX(C_{v,d}, v, d)$
17     end
18 end
19 $\mathbb{S} \leftarrow$ all nodes in the network $\setminus\{d\}$ ordered by $cost$
20 forall the $v \in \mathbb{S}$ do
21     $C_{v,d} \leftarrow \underset{T \in C_{v,d}, |T| \le ncand}{\arg\min} EAX(T, v, d)$
22 end
```

its candidates to their CS and $minnode$ is removed from $\mathbb{S}$. This process is done by means of the function *merge*, which combines both CSs and order them in increasing order of their cost (EAX). Note that proceeding this way, MTS finishes in $N - 1$ iterations, where N is the number of nodes in the network.

In the description of MTS given above, the optimal CSs for all the nodes in the network are computed assuming there is not any limitation in the number of candidates, as proposed in the original version of this algorithm.

In order to limit the maximum number of candidates, maintaining the optimality of the algorithm, we have added the lines 19–22. Here the nodes are visited in increasing order of their cost, and an exhaustive search is done over all subsets of the CSs with cardinality $\le ncand$. Since MTS first find the optimal CSs in the case of *infinite candidates* (i.e. all possible nodes can be selected as candidates), and then we look for the best subset of CSs with at most $ncand$ elements, the final Canditates Sets will be the optimal Canditates Sets.

Like the previous algorithms, we apply MTS in the example of Fig. 11.7 to find the CSs. According to MTS algorithm $\mathbb{S} = \{S, A, B\}$. The cost of nodes S, A and B to the destination D is $\frac{1}{0.15}$, $\frac{1}{0.4}$ and $\frac{1}{0.31}$, respectively. The result of each iteration of the MTS algorithm is shown in Table 11.5, where the first item in each cell is the current best CS for the corresponding node, and the second item is the current smallest expected number transmissions using that set. In the first iteration since the

Table 11.5 Candidates Set selection for the Fig. 11.7 using MTS

Iteration	S	A	B
1	$\{D, A\}$, 3.36	–	$\{D, A\}$, 2.79
2	$\{D, A, B\}$, 3.22	–	–

Table 11.6 Candidates set and EAX of each node in Fig. 11.7 using different algorithms

Node	ExOR	OAPF	LCOR or MTS
S	{A, B}, 3.64	{D, B}, 3.46	{D, A}, 3.36
A	{D}, 2.50	{D}, 2.50	{D}, 2.50
B	{D, A}, 2.79	{D, A}, 2.79	{D, A}, 2.79

EAX of A is the minimum ($EAX(\{D\}, A, D) = 2.5$), it removes from $\mathbb{S}$. Then, MTS adds A and its candidates ($C_{A,D} = \{D\}$) to the CSs of all neighbors of A, i.e., nodes S and B (see iteration 1 in Table 11.5). Note that in each iteration the EAX of each node is updated according to the new CS. In the second iteration the node with the minimum EAX is B ($EAX(\{D, A\}, B, D) = 2.79$); it is removed from $\mathbb{S}$ and B are added to the CSs of all neighbors of B which are still in $\mathbb{S}$, i.e., node S. Now, each node has a set of candidates to reach D. Note that, until this step there is not any limitation on the number of candidates. Doing an excursive search with constraint $ncand = 2$ over the sets which are found by the original version of MTS results in the optimum CS with length at most equal to 2.

We summarize the CSs and EAX of each node in Fig. 11.7 using different algorithms under study in Table 11.6. The first item in each cell is the CS for the corresponding node, and the second item is the EAX of the corresponding node using the said set. As we can see in Table 11.6 the algorithms that use EAX to select CSs have better expected number of transmissions than ExOR which uses ETX for selection of candidates.

11.6 Network Coding Opportunistic Routing

Mac-independent Opportunistic Routing & Encoding (MORE) [10] is an OR protocol that can be used in both unicast and multicast scenarios. It deploys the advantages of NC to improve performance of OR in wireless multicast networks. Duplicate transmissions are avoided by randomly mixing packets before forwarding. The sender creates a linear combinations of packets and broadcasts the resulting packet after adding a MORE header containing the CS. Each receiving node discards the packet if it is not linearly independent from the other packets received before, or if its ID does not appear in the candidates list. Otherwise, it linearly combines the received coded packets and rebroadcasts the new packet.

COPE [21, 22] is a practical NC mechanism for supporting efficient unicast communication in a wireless mesh network. It employs opportunistic listening to enable each node to learn local state information and encoded packet broadcasting to improve the network throughput. It exploits the shared nature of the wireless medium which broadcasts each packet in a small neighborhood around its path. Each node stores the overheard packets for a short time [21]. It also tells its neighbors which packets it has heard by annotating the packets it sends. When a node transmits a packet, it uses its knowledge of what its neighbors have heard to perform opportunistic coding; the node XORs multiple packets and transmits them as a single packet if each intended next-hop has enough information to decode the encoded packet. Motivated by COPE, several other coding-aware routing mechanisms have been proposed [29, 44], which are aimed at improving the network throughput by combining routing with inter-flow NC.

Coding-aware Opportunistic Routing mechanism & Encoding (CORE) [45, 46] is a coding-aware OR mechanism that combines Opportunistic Routing and localized inter-flow NC for improving the throughput performance of a MWN. Through OR, CORE allows the next-hop node with the most coding gain to continue the packet forwarding. Through localized NC, CORE attempts to maximize the number of packets that can be carried in a single transmission. When a node has a packet to send, it simply broadcasts the packet, possibly encoded with other packet(s), which may be received by some of the candidates in its CS. The candidates receiving the packet collaborate to select the best candidate among them in a localized manner, which is the one with the most coding opportunities. This forwarding process is repeated until the packet reaches its intended destination. In CORE, geo-distance metric and timer-based coordination have been used to select and coordinate the candidates, respectively.

11.7 Geographic Opportunistic Routing

Geographic Random Forwarding (GeRaF) [53] is a forwarding protocol which selects a set of candidates and prioritizes them using geographical location information. Only those neighboring nodes closer to the destination than the sender can be candidates. The priority of selected candidates is based on their geo-distances to the destination. The CS and prioritization can easily be implemented via an RTS-CTS dialog at the MAC layer, which also ensures that a single forwarder can be chosen.

Geographic Opportunistic Routing (GOR) [50] is used in geographic routing scenarios and adopts timer-based coordination with local candidates order. Authors showed that giving the nodes closer to the destination higher priority is not always the optimal way to achieve the best throughput. They proposed a local metric named Expected One-hop Throughput (EOT) to characterize the local behavior of GOR in terms of bit-meter advancement per second. Based on EOT, which considers the coordination overhead, they proposed a candidate selection scheme.

Yang et al. [47] proposed a protocol called Position based Opportunistic Routing (POR). In POR, when a source wants to send data packet to the destination, it finds its CS according to the distance between its neighbors and the destination. The neighbor which the nearest to the destination will have the highest priority. They fixed the maximum number of candidates in each node to 5. When a candidate receives a packet, it checks its position in the CS and waits for some time slots to forward the packet. If it hears the same packet being sent by the other nodes, it will simply discard the packet.

We defined a new metric for candidate selection based on the expected distance progress of sending a packet under Opportunistic Routing in [12]. By using this metric we proposed a hop-by-hop candidate selection and prioritization algorithm that we call Distance Progress Opportunistic Routing (DPOR). In contrast to other algorithms, in DPOR each node selects its candidates set independently, without considering the other nodes' candidates sets. DPOR only relies on the neighbors' geographic position of forwarder and the links delivery probability between forwarder and its neighbors.

11.8 Multicast Opportunistic Routing

Multicast is an important communications paradigm in wireless networks. The availability of multiple destinations can make the selection of CSs and the coordination between candidates complicated. There are few works that have tried made to adapt OR for multicast scenarios.

In [23] the source first creates the shortest path tree to reach all destinations based on the ETX of each link. Then the nodes not only receive packets from their father in the tree, but also can overhear packets from its sibling nodes. It uses random linear NC to improve multicast efficiency and simplify node coordination.

The authors in [41] used a Steiner tree based on ETX and data packets were forwarded through the links using OR. Their protocol constrains the nodes involved in routing a packet to be near the default multicast tree. The average EAX of each candidate to reach a sub-group of destinations is used as the cost of reaching to multiple destinations.

In a recent work, Le and Liu [30] propose Minimum Steiner Tree with Opportunistic Routing (MSTOR) which is an overlay multicast to adapt OR in wireless network. They construct a minimum overlay Steiner tree, and map it into unicast OR relay path connecting the source with all destinations. Their protocol does not exploit opportunistic receptions across different links in that tree. They employed unicast OR on each link of the tree.

In [11], we propose a new multicast routing protocol based on OR named Multicast Opportunistic Routing Protocol (MORP0. It opportunistically employs a set of forwarders to send a packet toward all destinations. MORP uses a three-way-handshaking where the sending node selects the candidates and towards which destinations they have to forward the packet. The basic idea of MORP is to form a CS to reach the destinations and based on the candidates which successfully receive the packet, selects a set of candidates as the forwarders to reach all destinations. Each forwarder is responsible for sending the packet to a subset of destinations. Indeed, based on the candidates that successfully receive the packet in each transmission, MORP builds a multicast tree on the fly using OR and forwards the packet through the tree.

11.9 Analytical Models of Opportunistic Routing

There are some papers which propose analytical models to study the performance of OR. Baccelli et al. [3] used simulations to show that Opportunistic Routing protocols significantly improve the performance of MWNs compared to the shortest path routing algorithms, and elaborated a mathematical framework to prove some of the observations obtained by the simulations.

In [35] an analytical approach for studying OR in wireless multi-hop networks have been proposed. They used lognormal shadowing and Rayleigh fading models for packet reception. In their model they assume that the nodes are uniformly distributed over the plane. The authors did not consider any specific candidate selection algorithm, but simply compute the expected progress of the packet transmissions based on the probability of any node in the progressing region successfully receives the packet.

Zubow et al. in [24] claimed that shadow fading losses for spatially close candidates are not independent from each other, unlike commonly assumed. They presented measurements obtained from an indoor testbed and concluded that correlations can not be neglected if nodes are separated by less than 2 m. The authors of [43] proposed an utility-based model for OR and claimed that for the optimal solution it is necessary to search all loop-free routes from the source to the destination. They proposed both optimal and heuristic solutions for selecting the candidates according to their utility function. In [34] an algebraic approach is applied to study the interaction of OR routing algorithms and routing metrics. They showed that Opportunistic Routing in combination with ETX could degrade the performance of network.

In [9] we proposed a Markov model to assess the improvement that may be achieved using OR. This model is the basis of the analytic approach described in Sect. 11.10.1. At the same time, Li and Zhang published an analytical framework to estimate the transmission costs of packet forwarding in wireless networks [32]. Both approaches are similar in their formulation, although differ in the way the model is solved: our model leads to a discrete phase-type distribution, while in [32] transmission costs are computed using spectral graph theory. In [8] we have derived the equations that yield the distances of the candidates in OR such that the per transmission progress towards the destination is maximized. There, we have proposed a lower bound to the expected number of transmissions needed to send a packet using OR.

In [7], the issue of optimal CS selection in the OR has been addressed. They provide an analytical framework to model the problem of selecting the optimal CS for both the constrained and unconstrained CS selection. They proposed two algorithms for optimal CS selection, one for the constrained and one for the unconstrained case.

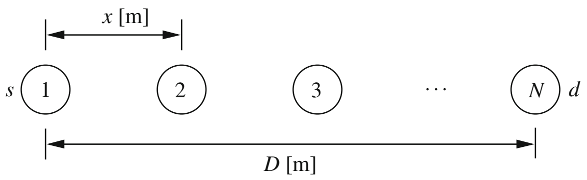


Fig. 11.8 Linear network topology

11.10 Performance Evaluation

In this section we propose a simple Markov chain model to study the performance of the candidate selection algorithms described in Sect. 11.5. Our model can be used to compute the probability distribution and moments of the number of transmissions needed to send a packet from the source to the destination in a variety of scenarios. For each node, the ordered list of candidates and the delivery probability to each of them are inputs to our model. Hence, our model does not require any specific assumptions about the network topology nor the mechanism for selection and prioritization of candidates.

11.10.1 Markov Model

We will consider one *tagged* connection. Each node has a set of candidates that can opportunistically route the packets towards the destination. In order to simplify the explanation of our model, we will first describe a simple scenario, and then we will generalize it.

Consider a linear network topology of N nodes equally spaced a distance $x = D/(N-1)$, being D the distance between the source s and the destination d of the tagged connection (see Fig. 11.8).

Let $p(x)$ be the probability of successfully delivering a packet to node located at a distance x. The nodes retransmit the packets until successful delivery. With the assumption of independent delivery probabilities, and the nodes always routing the packets to their closest neighbor, the average number of transmissions N_t in uni-path routing is given by:

$$N_t = \frac{N-1}{p(x)}. \qquad (11.2)$$

Assume now that OR is used with a list of 2 candidates. That is, we assume that upon transmission, if any of the next 2 neighbors toward the destination receive the packet, the closest node to the destination opportunistically becomes the next-hop towards the destination. We can model this routing by means of the absorbing DTMC depicted in Fig. 11.9. The transition probabilities are given by:

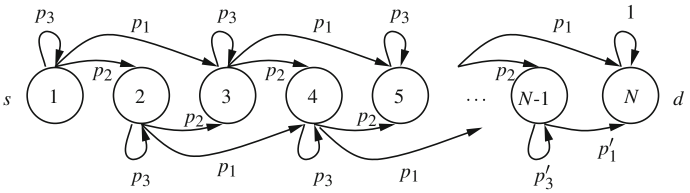


Fig. 11.9 Opportunistic Routing model with 2 candidates

$$\begin{aligned}
p_1 &= p(2x) \\
p_2 &= p(x)\,(1 - p_1) \\
p_3 &= 1 - (p_1 + p_2) \\
p'_1 &= p(x) \\
p'_3 &= 1 - p'_1.
\end{aligned} \tag{11.3}$$

The DTMC models the progress of a packet from source to destination. The initial state is 1 and a transition occurs at each transmission shot. When the DTMC is in state i it represents that the packet has progressed up to node i. Eventually the packet reaches its destination when the DTMC is absorbed at state N.

A similar DTMC can be easily derived for 3 candidates and so on, until all possible nodes are candidates (we shall refer to this case as *infinite candidates*). Furthermore, the model can be readily extended to an arbitrary network. The only ingredients needed to build the transition probability matrix are the CSs involved in the routing from s to d, and the delivery probabilities to reach them. Notice that these CSs are: the candidates of node s towards d, the candidates of these candidates towards d, and so on until d (whose CS is the empty set). A detailed description of the process to build to build the transition probability matrix is given next.

We will utilize graph theory notation for the sake of being concise. Let $G = (V, E)$ be the graph of the network. The vertex s is the source and d is the destination of the tagged connection ($s, d \in V$). Note that we will use node/vertex and link/edge interchangeably. Let $p(i, j) > 0$ the delivery probability of the edge between the pair of vertices i, j. Let $C_{i,d} = \{c_i(1), c_i(2), \cdots c_i(n_i)\}$, $C_{i,d} \subseteq V$ the ordered set of candidates of vertex i ($c_i(1)$ is the best candidate to reach d and $c_i(n_i)$ is the worst). As before, each vertex of the graph is a state of the DTMC, being d the absorbing state. The transition probabilities $p_{ij} \neq 0$ are given by:

$$p_{ij} = p(i, j), \qquad i \neq d,\ j = c_i(1) \tag{11.4}$$

$$p_{ij} = p(i, j) \prod_{l=1}^{k-1} (1 - p(i, c_i(l))), \qquad i \neq d,\ j = c_i(k),\ k = 2, \cdots, n_i \tag{11.5}$$

$$p_{ii} = 1 - \sum_{l \in C_{i,d}} p_{il} = \prod_{l=1}^{n_i} (1 - p(i, c_i(l))), \quad i \neq d \tag{11.6}$$

$$p_{ii} = 1, \qquad i = d\,. \tag{11.7}$$

Note that the two expressions given for p_{ii} in Eq. (11.6) follow from the stochastic nature of the transition matrix (the first one), and because p_{ii} is the probability that none of the candidates ($C_{i,d}$) receives the packet (the second one).

Without loss of generality, we can number the nodes such that the source and the destination are respectively 1 and N, and for any node i, its candidates satisfy: $c_i > i \;\; \forall c_i \in C_{i,d}$. Note that, neglecting self-transitions, the former condition implies that the graph is loop free. This condition holds assuming that the candidate selection algorithm uses some kind of strict order, i.e., for a node j to be included into the set of candidates of i it must be strictly *closer* to the destination than i. Hence, a loop $i = j_0 \rightarrow j_1 \rightarrow j_2 \rightarrow \cdots \rightarrow j_n = i$ (by transitivity) would imply that i is strictly closer to the destination than i, which is a contradiction. This is an obvious assumption for a well designed candidate selection algorithm. Otherwise, a node i would choose as candidate a node having a larger cost to reach the destination than the node i itself. With these assumptions, the transition matrix of the resulting chain has the triangular form:

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1N} \\ 0 & p_{22} & p_{23} & \cdots & p_{2N} \\ 0 & 0 & p_{33} & \cdots & p_{3N} \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{T} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}. \tag{11.8}$$

where $\mathbf{T}$ governs the transmissions before reaching the destination, and $\mathbf{t} = [p_{1N}\; p_{2N} \; \cdots \; p_{N-1N}]^{\mathrm{T}}$ are the probabilities to reach the destination in one transmission from the nodes $1, \cdots, N-1$.

Let X_1 be the random variable equal to the number of transitions from the source (node 1) until absorption. Note that in our model this is the number of transmissions since the source first transmits the packet, until it is received by the destination. The DTMC obtained in our model represents a discrete phase-type distribution [27]. Thus, the point probabilities and factorial moments of X_1 are given by:

$$P\{X_1 = n\} = \boldsymbol{\tau}\, \mathbf{T}^{n-1}\, \mathbf{t}, \;\; n \geq 1 \tag{11.9}$$

$$\mathrm{E}[X_1(X_1-1)\cdots(X_1-k+1)] = n!\, \boldsymbol{\tau}\, (\mathbf{I} - \mathbf{T})^{-n}\, \mathbf{T}^{n-1}\, \mathbf{1} \tag{11.10}$$

where we define $\boldsymbol{\tau} = \begin{bmatrix} 1 \; 0 \cdots 0 \end{bmatrix}^{\mathrm{T}}$ and $\mathbf{1}$ a column vector of 1s. Note that $\mathbf{T}$, and thus also $\mathbf{I} - \mathbf{T}$, are triangular matrices, which simplifies the computation of their inverses.

11.10.2 Expected Number of Transmissions

If we are only interested on the expected number of transmissions, we can derive a recursive equation as follows. Let X_i $(i \neq d)$ be the random variable equal to the number of transitions from the state i until absorption. Clearly:

$$\mathrm{e}[X_i] = 1 + \sum_{j \in C_{i,d}} p_{ij}\, \mathrm{e}[X_j] = 1 + p_{ii}\, \mathrm{e}[X_i] + \sum_{l=1}^{n_i} p_{il}\, \mathrm{e}[X_l]$$

grouping $\mathrm{e}[X_i]$ we get:

$$\mathrm{e}[X_i] = \frac{1 + \sum_{l=1}^{n_i} p_{il}\, \mathrm{e}[X_l]}{1 - p_{ii}}, \quad i \neq d\,. \tag{11.11}$$

Taking $\mathrm{e}[X_d] = 0$, the Eq. (11.11) can be used to compute the expected number of transmissions needed to send a packet from the source s to the destination d by using $C_{s,d}$ as the CS of s to reach node d. Note that the loop free property of the chain guarantees that the recursive Eq. (11.11) is finite.

Equation (11.11) has been obtained by other methods in [16] and [52], where it is referred to as *least cost Any-Path* and *Expected Any-Path Transmissions* (EAX) respectively. As explained in Sect. 11.5, some candidate selection algorithms for OR use the ETX metric. Although ETX is much simple to compute than EAX, it does not accurately compute the expected number of transmissions under Opportunistic Routing.

11.10.3 Propagation Model

In order to assess the delivery probabilities we will assume that the channel impediments are characterized by a shadowing propagation model: the power received at a distance x is given by:

$$P_r(x)|_{dB} = 10\, \log_{10}\left(\frac{P_t\, G_t\, G_r\, \lambda^2}{L\,(4\,\pi)^2\, x^\beta}\right) + X_{dB} \tag{11.12}$$

where P_t is the transmitted power, G_t and G_r are the transmission and reception antenna gains respectively, L is a system loss, λ is the signal wavelength (c/f, with $c = 3 \times 10^8$ m/s), β is a path-loss exponent and X_{dB} is a Gaussian random variable with zero mean and standard deviation σ_{dB}.

Packets are correctly delivered if the received power is greater than or equal to `RXThresh`. Note that we shall not consider collisions in our model. Thus, the delivery probability at a distance x ($p(x)$) is given by:

Table 11.7 Default values in *ns-2* for the shadowing propagation model

Parameter	Value
P_t	0.28183815 W
`RXThresh`	3.652×10^{-10} W
G_t, G_r, L	1
f	914 MHz

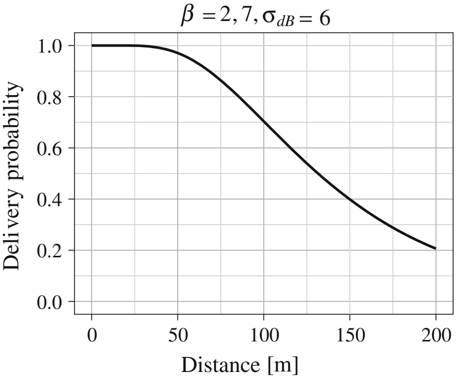


Fig. 11.10 Delivery probability versus distance for a path loss exponent $\beta = 2, 7$ and standard deviation $\sigma_{dB} = 6$ dBs

$$p(x) = Prob(P_r(x)|_{dB} \geq 10\ \log_{10}(\texttt{RXThresh})) = Q\left(\frac{1}{\sigma_{dB}} 10\ \log_{10}\left(\frac{\texttt{RXThresh}\ L\ (4\,\pi)^2\, x^{\beta}}{P_t\ G_t\ G_r\ \lambda^2}\right)\right) \tag{11.13}$$

where $Q(z) = \frac{1}{\sqrt{2\pi}} \int_z^{\infty} \mathrm{e}^{-y^2/2} dy$.

We have set the model parameters to the default values used by the network simulator (ns-2) [40], given in Table 11.7. We shall use these values in the numerical results presented in Sect. 11.10. With these parameters the link delivery probability is approximately 40 % at the distance of 150 m (Fig. 11.10).

11.10.4 Evaluation Methodology

We shall use the notation ExOR(n) to refer to ExOR with $ncand = n$, and similarly for the other algorithms under study (see the legend of Fig. 11.11, 11.12, 11.13, 11.14, 11.15, 11.16, 11.17, 11.18, and 11.19).

We have proceed as follows:

- First the network topology is set up randomly placing the nodes in a square field with diagonal $D = 300$ m, except the source and the destination which are placed at the end points of one of the diagonals. We consider scenarios with different number of nodes ($10 \leq N \leq 50$).

- The shadowing propagation model described in Sect. 11.10.3 is used to assess the delivery probabilities of the links (q_{ij} in the model described in Sect. 11.10.1). We have assumed that a link between any two nodes exists only if the delivery probability between them is greater (or equal) than $min.dp = 0.1$.
- We assume that the topology and delivery probabilities are known by all nodes, and for each of them it is used one of the algorithms described in Sect. 11.5 to compute the CSs for the given destination.
- Finally, we use the DTMC model described in Sect. 11.10.1 to compute the following performance measures: expected number of transmissions, variance of the expected number transmissions, probability of the number of transmissions.

We have done this evaluation using the R numerical tool [36]. Each point in the plots is an average over 100 runs with different random node positions. We have used this methodology for each of the algorithms described in Sect. 11.10.1, and for a different maximum number of candidates: $ncand = 2, 3, 4, 5, \infty$. Recall that we refer as $ncand = \infty$ to the case when there is no limit on the maximum number of candidates and all possible nodes can be selected as candidates.

As an estimation of the computational cost of the algorithms, we have measured the execution time it takes to compute the CSs in each scenario. These times have been obtained running the algorithms on a computer with an Intel Xeon Dual-Core 2 2.33 GHz, FSB 1333 MHz, with 4 MB cache and 12 GB of memory.

11.10.5 Numerical Results

11.10.5.1 Expected Number of Transmissions

First, we examine in detail the case with at most 3 candidates for each node ($ncand = 3$), as shown in Fig. 11.11. For the sake of comparison, we have included the scenarios using uni-path routing and also the optimal candidate selection algorithm in the case $ncand = \infty$ (we shall refer to it as Opt(∞)). Note that uni-path routing is equivalent to use $ncand = 1$ in any of the OR algorithms under study. The curves have been obtained varying the number of nodes, but maintaining the distance $D = 300$ m between the source and the destination, thus, increasing the density of the network.

As a first observation in Fig. 11.11, we can see that using any OR algorithm outperforms the traditional uni-path routing. Regarding the optimal algorithms, LCOR and MTS, we have validated that they choose exactly the same CSs, and thus, the curves are the same. Additionally, for $ncand = \infty$ the expected number of transmissions for LCOR and MTS are the same, so we show only one of the curves obtained with LCOR(∞) and MTS(∞) (indicated as Opt(∞)).

We can see that the expected number of transmissions obtained with OAPF is only slightly larger than those obtained with the optimal algorithms. Finally, we observe that the expected number of transmissions required by ExOR is significantly larger

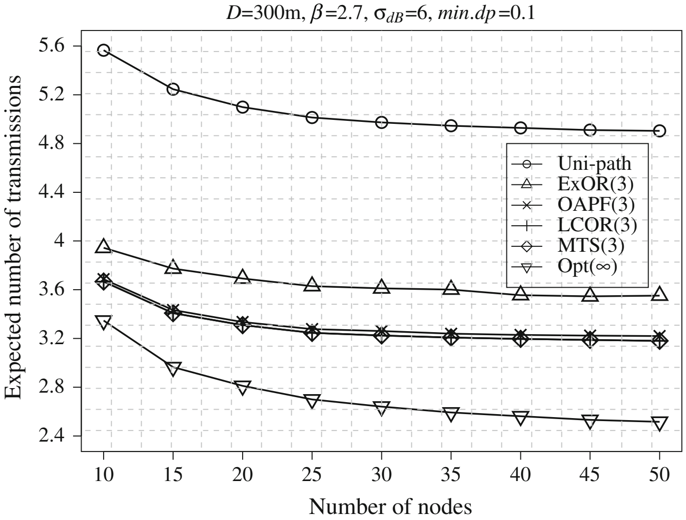


Fig. 11.11 Expected number of transmissions in the case $ncand = 3$

than any other OR algorithms. The reasons that motivate this inferior performance of ExOR are the following: recall that ExOR is a simple algorithm that uses ETX as the metric for selecting candidates. It looks for the candidates running SPF after removing the links to the nodes that have already been selected as candidates. By doing this, the candidates tend to be chosen close to each other. In [8] we have investigated the optimal position of the candidates and we have shown that they are not clustered, but distributed over distances that approximate to the destination. Therefore, we conclude that ExOR does a coarse selection of the CS. On the other hand, recall that OAPF incrementally adds the nodes to the CS that are most effective at reducing the EAX. Although this does not guarantee choosing the optimal CSs, we can see from the Fig. 11.11 that the results are very close to the optimal algorithm.

Regarding the scenario with $ncand = \infty$, Fig. 11.11 shows that it achieves a noticeable reduction of the expected number of transmissions compared to the scenario with $ncand = 3$. However, as shown in Fig. 11.12, this is at cost of using a large number of candidates. Note that implementing an OR protocol with a high number of candidates is difficult, and possibly will introduce large signaling overhead and duplicate transmissions. Therefore, the differences obtained with $ncand = 3$ and $ncand = \infty$ in a real scenario, are likely to be much smaller than those shown in Fig. 11.11.

For other scenarios we have obtained similar results. For instance, Figs. 11.13 and 11.14 have been obtained, respectively, maintaining the total number of nodes equal to $N = 10$ and $N = 50$ (thus, representing a low and high density network), and varying the maximum number of candidates to: $ncand = 1, 2, \cdots, 5$ and ∞. Note that $ncand = 1$ is equivalent to uni-path routing, thus, the expected number of transmissions obtained for $ncand = 1$ is the same for all algorithms.

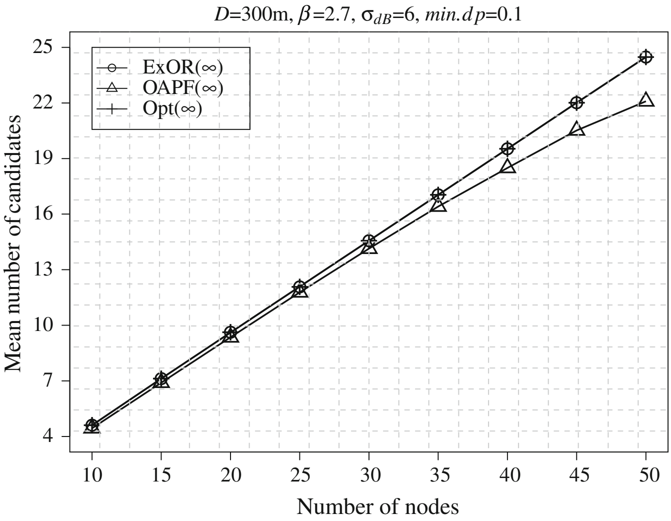


Fig. 11.12 Mean number of candidates in the case $ncand = \infty$

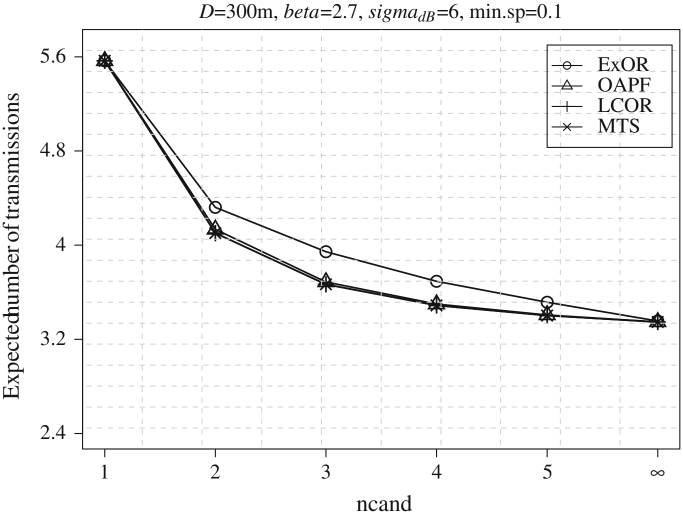


Fig. 11.13 Expected number of transmissions for the random topology with $N = 10$ nodes varying the maximum number of candidates

In the case of $ncand = \infty$ all algorithms have almost the same expected number of transmissions. This comes from the fact that in this case there is not any limitation on the maximum number of candidates. Therefore, all nodes which are closer to the destination than the source can be selected as candidates, and all of the algorithms have almost the same CSs. Note that since ExOR uses ETX as the metric to select

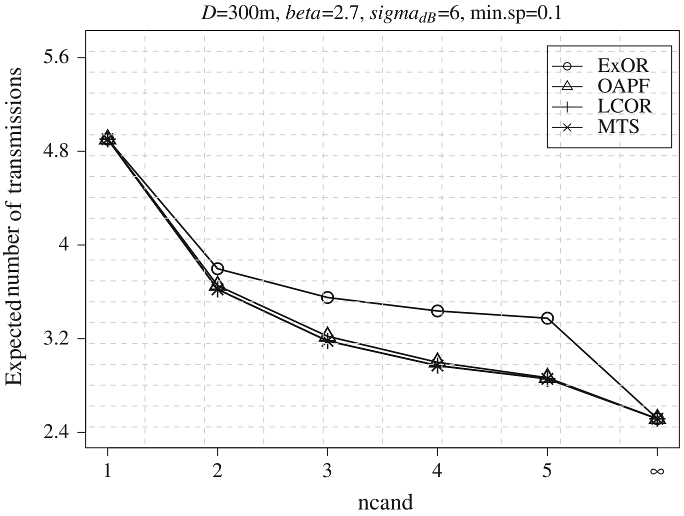


Fig. 11.14 Expected number of transmissions for the random topology with $N = 50$ nodes varying maximum the number of candidates

candidates, the order of candidates may be different compared with the CSs in the other algorithms. Because of that the expected number of transmissions in the case of ExOR with $ncand = \infty$ has a very small difference compared with the other algorithms (not noticeable in the graphs).

By comparing Figs. 11.13 and 11.14 we can see that the difference between ExOR and the other algorithms is higher in a dense network ($N = 50$). This comes from the fact that in a dense network there is a larger number of possible choices of the CSs. Thus, limiting the maximum number of candidates makes the selection of the candidates sets more critical. However, we can see that the difference between OAPF and the optimal algorithms is kept small even in a dense network. We can see that increasing maximum number of candidates ($ncand$) from 1 to 2 results in an important gain in all cases and increasing $ncand$ from 5 to ∞ is more important in the dense topology.

11.10.5.2 Variance of Expected Number of Transmissions

One of the metrics which can also be calculated with our model is the variance of expected number of transmissions from source to the destination. Figures 11.15 and 11.16 show the variance of expected number of transmissions for a low ($N = 10$) and high ($N = 50$) density network, respectively. Since with one candidate all algorithms have the same result as uni-path routing the variances of the expected number of transmissions for $ncand = 1$ are the same.

Figures 11.15 and 11.16 show that using OR the variance of the expected number of transmissions is significantly reduced compared with uni-path routing. It is also

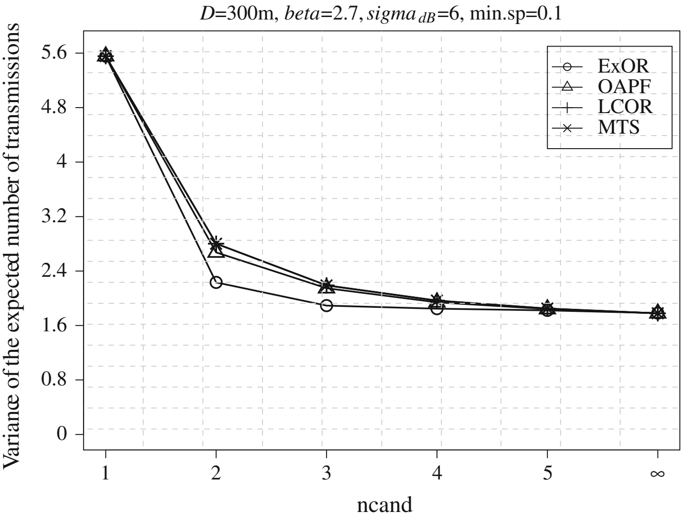


Fig. 11.15 Variance of the expected number of transmissions for the random topology with $N = 10$ nodes varying the maximum number of candidates

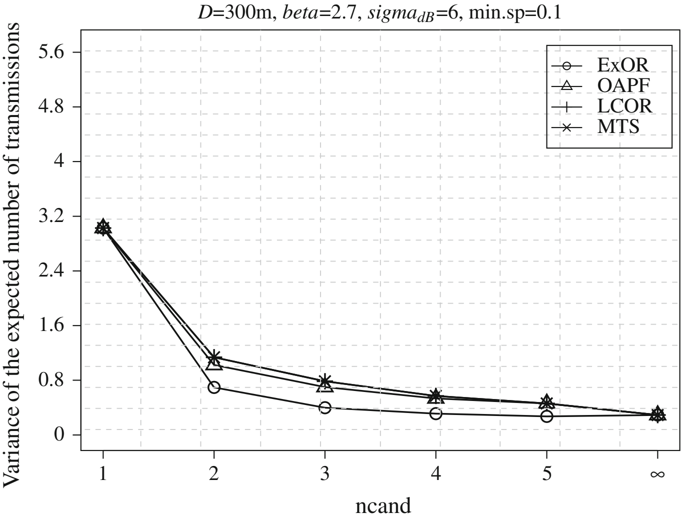


Fig. 11.16 Variance of the expected number of transmissions for the random topology with $N = 50$ nodes varying the maximum number of candidates

observed that while the variance decreases with the value of *ncand*, just a small of value (typically 2 or 3 candidates per node) is enough to attain a significant part of the potential reduction. This effect is even more noticeable when the candidate selection algorithm employed is ExOR. Furthermore, while ExOR is the algorithm that yields the highest mean number of transmissions, as it was shown above, it achieves the lowest variance.

The reduction of variance of the expected number of transmissions, compared with uni-path routing, has two important benefits. Firstly, the variability of the transmission delays may be significantly reduced using OR. Secondly, this fact indicates that the number of retransmissions of a packet by the same node may be also reduced using OR. This may also contribute on the reduction of the transmission delay variability, due to the back-off algorithm used at the MAC layer.

11.10.5.3 Probability Distribution of the Number of Transmissions

For having a more detailed comparison, we have included the probability distribution of the number of transmissions for $ncand = 1$, 3 and ∞ for a small number of nodes $N = 10$ and a large one $N = 50$, in Figs. 11.17 and 11.18, respectively.

The probability curves for the $ncand = 1$ case (uni-path routing) in both Figs. 11.17 and 11.18 are almost the same. These figures show that for $N = 10$ in the uni-path routing, about 14 % of packets reach the destination with 3 transmissions, while about 40 % of packets need 6 or more transmissions. In Figs. 11.17 and 11.18 we can see that, by using OR algorithms, the number of transmissions needed to reach the destination is significantly reduced with respect to the uni-path routing approach. The curves for all algorithms except ExOR are almost the same. In a low density network ($N = 10$), using the optimal candidate selection algorithms (LCOR or MTS) in the $ncand = 3$ case, 18 and 37 % of packets reach the destination with 2 and 3 transmissions, respectively, while using ExOR only about 5 % of

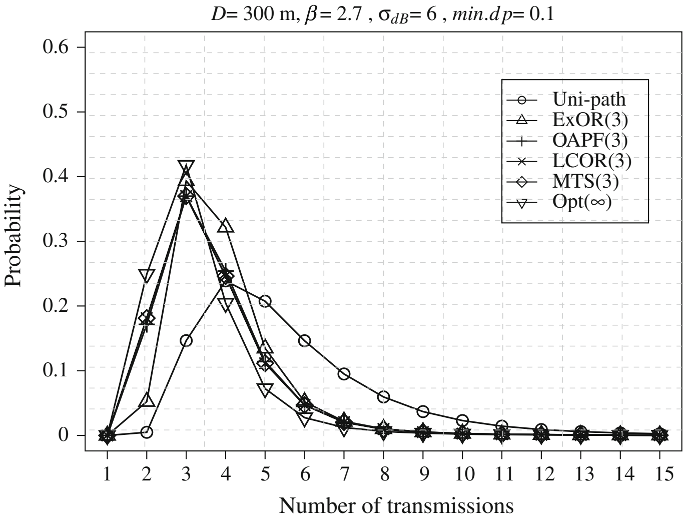


Fig. 11.17 Probability of the number of transmissions for the random topology with $N = 10$ nodes

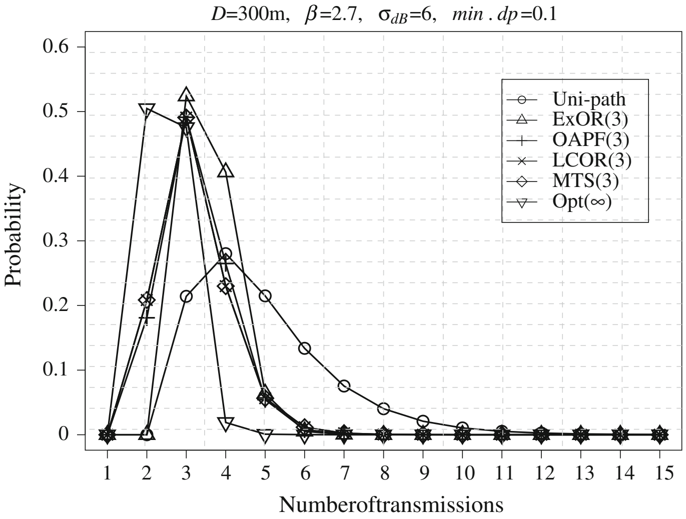


Fig. 11.18 Probability of the number of transmissions for the random topology with $N = 50$ nodes

packets reach the destination with 2 transmissions. In the network with more nodes ($N = 50$), LCOR, MTS and even OAPF can select the candidates which are close to the destination. Therefore as we can see in Fig. 11.18 by using these algorithms with $ncand = 3$ about 20 and 50 % of packets reach the destination with 2 and 3 transmissions, respectively.

By comparing the Figs. 11.17 and 11.18 we can see that the probabilities change significantly for the $ncand = \infty$ case. For instance, in Fig. 11.18 about 50 % of packets reach the destination only with 2 transmissions, while in the low dense network ($N = 10$) only 25 % of packets reach the destination with 2 transmissions. Looking at Fig. 11.12 we can see that, the $ncand = \infty$ case uses 25 candidates in a dense network ($N = 50$). With such a large number of candidates it is likely that some candidate close to the destination will receive the packet, thus, allowing the delivery to the destination with only two transmissions.

11.10.5.4 Execution Time

In this section we estimate the computational cost of the algorithms under study by measuring the *execution time* it takes to compute the CSs towards the destination necessary to solve the DTMC model described in Sect. 11.10.1. Recall that these CSs are: the candidates of the source s towards the destination d, the candidates of these candidates towards d, and so on until d (whose candidates is the empty set). Notice that for ExOR this requires calling Algorithm 1 for the source s, for its candidates, the candidates of these candidates, and so on until d. For the other algorithms, computing the CSs of the source requires the computation of all the necessary CSs. This comes

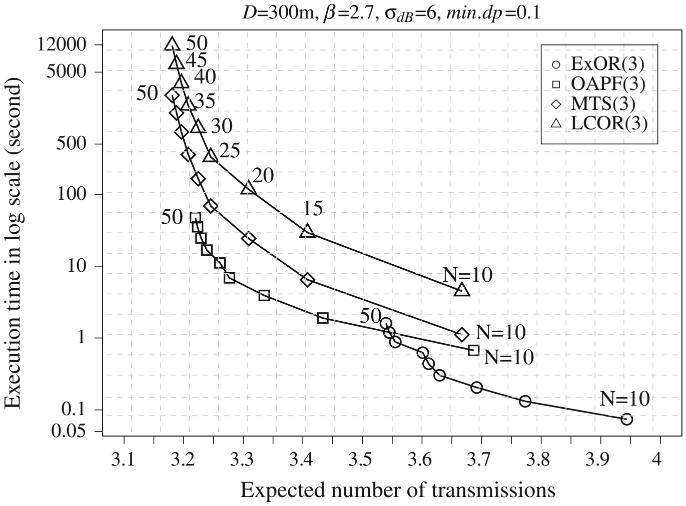


Fig. 11.19 Expected number of transmissions and execution time of all algorithms

from the fact that the other algorithms are based on the EAX metric, which requires the CSs. Therefore, for the algorithms OAPF, LCOR and MTS, the execution time is the time it takes calling only once the Algorithms 2, 3 and 4, respectively.

Figure 11.19 shows the expected number of transmissions versus the execution time in logarithmic scale. We have selected $ncand = 3$ as a sample case for our study. So, the points in Fig. 11.19 have been obtained by averaging over the 100 runs of the corresponding points in Fig. 11.11. The values next to the points represent the number of nodes of the network N.

We can see that for all the algorithms, the larger is the number of nodes the lower is the expected number of transmissions and the higher is the execution time. As expected, the fastest algorithm is ExOR whereas LCOR is the slowest. For instance, when the number of nodes in the network is 50, LCOR needs about 3.3 h to finish. Obviously, with a maximum number of candidates larger than 3 the execution time will be much longer. OAPF lies between the exhaustive search of the optimal algorithms and the simplicity of ExOR, and thus, has an execution time that falls in between these algorithms, e.g. 0.6–47 s for the low and high density networks, respectively.

MTS and LCOR have the same expected number of transmissions while the execution time of MTS is much lower than LCOR. For instance in the high density network ($N = 50$) MTS needs about 40 min to finish while LCOR needs about 3.3 h. Recall that MTS(3) first looks for the optimal candidates sets without limiting the maximum number of candidates, and then the candidates sets are pruned to at most 3 elements. Therefore, the searching space for finding the optimal sets in MTS(3) is less than LCOR(3), which examines all the subsets of the neighbors of the nodes.

By comparing the two optimal algorithms that have been proposed in the literature, we can conclude that MTS outperforms LCOR in terms of the execution time.

Additionally, it is possible to obtain candidate selection algorithms, as OAPF, that have a performance close to the optimal algorithms with a much lower execution time. With simple algorithms as ExOR, the performance may be significantly poorer than the optimal.

11.11 Conclusions and Future Research Directions

11.11.1 Conclusions

In this chapter we have described the meaning of OR that has been introduced as a way of using the broadcast nature of MWNs. We have classified different research areas in OR: routing metrics, candidate section, candidate coordination, geographic OR and multicast OR. Then, we have surveyed the main research contributions in each category.

The two usual metrics that used in the literature of OR, ETX and EAX, have been discussed in detailed. Although ETX is simpler to compute than EAX, it does not accurately compute the expected number of transmissions under OR. The other important issue of OR is the candidate selection. We have described in detail four different candidate selection algorithms that have been proposed in the literature. They range from non-optimum, but simple, to optimum, but with a high computational cost. Regarding the different candidate coordination approaches we have described the four most used methods of coordination in OR: acknowledgment-based, timer-based, NC and RTS-CTS; and have explained the advantages and disadvantages of each approach.

Applying the OR paradigm to multicast routing is another new research direction. The availability of multiple destinations can make the selection of CS and coordination among them complicated. There are few works that have tried to adapt OR to multicast settings. We have briefly describe different protocols that apply OR to multicast routing. Most of them first create the shortest path tree to the destinations and then send the packet through the tree using OR.

After the general overview and introduction of OR we have focused on its performance analysis. First, we have surveyed the existing performance studies that are based on analytical models. Then, we have introduced our own contribution, a DTMC model to analyze the performance gain that may be achieved by using OR. In our model the nodes are represented by the states of the Discrete Time Markov Chain, and the state transitions model how the packet progresses through the network. The only ingredients needed to build the transition probability matrix are the candidates of each node, and the delivery probabilities to reach them. As a consequence, the proposed model can be applied independently of the candidate selection algorithm that is employed. The model leads to a discrete phase-type representation for the distribution of the number of transmissions that are needed to reach the destination node. An important advantage of the phase-type representation is that there exist simple and closed-form expressions for its distribution and moments.

We have applied our model to compare four relevant algorithms that have been described in the candidate selection section. We have compared different scenarios in terms of the expected, the variance and the probability of the number transmissions needed to send a packet from source to the destination. The algorithms have also been compared from the perspective of the execution time which is needed to construct the CSs.

Our numerical results have shown that using any OR algorithm outperforms the traditional uni-path routing. Furthermore, if the maximum number of candidates is not limited, all of the algorithms obtain almost the same expected number of transmissions. Such assumption is not realistic since the algorithms may choose a large number of candidates, which will introduce large signaling overhead and probably duplicate transmissions. When the maximum number of candidates is limited, our results have shown that the expected number of transmissions required by ExOR is larger than that of the other OR algorithms. This is because of the coarse selection of the CSs of ExOR, which relies on ETX. On the other hand, the performance obtained with OAPF has proven to be very close to the optimal algorithms. We have also observed that the variance of the number of transmissions can be substantially reduced by using OR. This result is specially important in networks with real-time requirements.

Regarding the execution times, the fact that ExOR is based on ETX makes this algorithm much faster than the others. For the optimum algorithms, we have observed that MTS outperforms LCOR. However, both algorithms require extremely large execution times to compute the CSs in a dense network (on the order of hours in a modern PC). On the other hand, OAPF is able to run the candidate selection with execution times orders of magnitude lower than the optimum algorithms (on the order of minutes) while its performance in terms of the expected number of transmissions was very close to the optimum algorithms. Therefore, we conclude that a fast and simple OR candidate selection algorithm such as OAPF may be preferable in dynamic networks, where the CSs are likely to be updated frequently.

11.11.2 Future Research Directions

To design an OR protocol two main issues should be considered: candidate selection and coordination. An efficient candidate selection algorithm selects a set of appropriate forwarders as the Canditates Set and gives the priority to each of them within a reasonable amount of time. To provide fast and efficient candidate selection algorithms with few topology information, more research in the candidate selection issue is needed.

Candidate coordination is another important issue in the design of Opportunistic Routing protocol which receives less attention than the candidate selection in the literature. Imperfect candidate coordination may cause duplicate transmissions which increases the control overhead and reduces the efficiency of the OR protocol. Candidate coordination requires reliable signaling among the nodes. A practical implementation of the candidate coordination algorithm needs to consider existing technologies. It can be implemented at the link or network layer. However a link layer implementation permits the design of an efficient signaling protocol. To implement an OR protocol in the link layer, the MAC layer IEEE 802.11 standard should be modified. Therefore, it needs more investigation to design a practical as well as an efficient candidate coordination mechanism.

Opportunistic Routing is usually investigated for wireless mesh networks where nodes do not have mobility. When dealing with mobile networks such as Ad-Hoc, Sensor networks and Vehicular Ad Hoc Network (VANET), OR is leading to include the mobility properties of these kind of networks. In the presence of mobility using an efficient and fast candidate selection algorithm is unavoidable.

Multicast is an important communications paradigm in wireless networks. Using OR for multicast delivery can improve the performance of multicast protocols. Furthermore, using OR approach for broadcasting a packet to all nodes in the networks can be another research direction.

The choice of metric has great impact on the performance of an OR protocol. Most of the works in the literature mainly focused on using ETX and EAX as primary metrics. Investigating the error of link delivery probability and its impact on the OR performance in different types of networks can be another research direction. Another potential direction for research is the routing metrics with various performance objectives, such as minimizing delay or maximizing energy efficiency.

Opportunistic Routing in multi-channel multi-radio networks can be an interesting research direction. It is interesting to investigate distributed algorithms to solve the channel assignment and candidate selection issues. In this issue, a sender may have different Canditates Sets on different channels.

Finally, security is another major concern in multihop wireless networks. It will be valuable to design secure OR protocols and integrate them into existing security framework to provide more robust and more secure information delivery service.

Acknowledgments This work was supported by the Spanish government and Generalitat de Catalunya through projects TIN2010-21378-C02-01, TIN2010-21378-C0202 and 2009-SGR-1167, respectively, and by the European Commission through the NoE EuroNF.

Glossary

COPE	Name used to identify the OR scheme using Network Coding proposed in [21]
CORE	Coding-aware Opportunistic Routing mechanism & Encoding
CS	Candidates Set
CTS	Clear-To-Send
DPOR	Distance Progress Opportunistic Routing
DTMC	Discrete Time Markov Chain
EAX	Expected Any-Path Transmission

EOT	Expected One-hop Throughput
ETX	Expected Transmission Count
ExOR	Extremely Opportunistic Routing
GeRaF	Geographic Random Forwarding
GOR	Geographic Opportunistic Routing
LCOR	Least-Cost Opportunistic Routing
NAV	Network Allocation Vector
NC	Network Coding
MORE	MAC-independent Opportunistic Routing & Encoding
MORP	Multicast Opportunistic Routing Protocol
MSTOR	Minimum Steiner Tree with Opportunistic Routing
MTS	Minimum Transmission Selection
OAPF	Opportunistic Any-Path Forwarding
OR	Opportunistic Routing
POR	Position based Opportunistic Routing
RTS	Request-To-Send
SDF	Selection Diversity Forwarding
SIFS	Short Interframe Space
SOAR	Simple Opportunistic Adaptive Routing
SPF	Shortest Path First
VANET	Vehicular Ad Hoc Network
MWN	Multi-hop Wireless Network

References

1. Aguayo D, Bicket J, Biswas S, Judd G, Morris R (2004) Link-level measurements from an 802.11b mesh network. SIGCOMM Comput Commun Rev 34:121–132. doi:10.1145/1030194.1015482
2. Ahlswede R, Cai N, Li SY, Yeung R (2000) Network information flow. IEEE Trans Inf Theor 46(4):1204–1216. doi:10.1109/18.850663
3. Baccelli F, Blaszczyszyn B, Muhlethaler P (2008) On the performance of time-space opportunistic routing in multihop mobile ad hoc networks. In: 6th International Symposium on Modeling and optimization in Mobile, Ad Hoc, and Wireless Networks and Workshops, 2008. WiOPT 2008. pp 307–316. doi:10.1109/WIOPT.2008.4586083
4. Biswas S, Morris R (2004) Opportunistic routing in multi-hop wireless networks. ACM SIGCOMM Comput Commun Rev 34(1):69–74
5. Biswas S, Morris R (2005) ExOR: opportunistic multi-hop routing for wireless networks. ACM SIGCOMM Comput Commun Rev 35(4):133–144
6. Bruno R, Nurchis M (2010) Survey on diversity based routing in wireless mesh networks: challenges and solutions. Comput Commun 33(3):269–282. doi:10.1016/j.comcom.2009.09.003
7. Cacciapuoti A, Caleffi M, Paura L (2010) Optimal constrained candidate selection for opportunistic routing. In: IEEE GLOBECOM, pp 1–5. doi:10.1109/GLOCOM.2010.5683490
8. Cerdà-Alabern L, Darehshoorzadeh A, Pla V (2010) On the maximum performance in opportunistic routing. In: 2010 IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM), pp 1–8. doi:10.1109/WOWMOM.2010.5534897

9. Cerdà-Alabern L, Pla V, Darehshoorzadeh A (2010) On the performance modeling of opportunistic routing. In: MobiOpp ’10: Proceedings of the Second International Workshop on Mobile Opportunistic Networking, ACM, New York, pp 15–21. doi:10.1145/1755743.1755747
10. Chachulski S, Jennings M, Katti S, Katabi D (2007) Trading structure for randomness in wireless opportunistic routing. In: SIGCOMM ’07: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications, ACM, New York, pp 169–180
11. Darehshoorzadeh A, Cerdà-Alabern L (2011) A new multicast opportunistic routing protocol for wireless mesh networks. In: Casares-Giner V, Manzoni P, Pont A (eds) NETWORKING 2011 workshops, lecture notes in computer science, vol. 6827, Springer, Berlin, pp 62–72
12. Darehshoorzadeh A, Cerdà-Alabern L (2012) Distance progress based opportunistic routing for wireless mesh networks. (Accepted). In: Wireless communications and mobile computing conference, IWCMC ’12. International
13. Darehshoorzadeh A, Cerdà-Alabern L, Pla V (2011) Modeling and comparison of candidate selection algorithms in opportunistic routing. Comput Netw 55(13):2886–2898. doi:10.1016/j.comnet.2011.06.009
14. De Couto DSJ, Aguayo D, Bicket J, Morris R (2005) A high-throughput path metric for multi-hop wireless routing. Wirel Netw 11(4):419–434
15. Dubois-Ferriere H, Grossglauser M, Vetterli M (2007) Least-cost opportunistic routing. In: Proceedings of 2007 allerton conference on communication, control, and computing
16. Dubois-Ferriere H, Grossglauser M, Vetterli M (2011) Valuable detours: least-cost anypath routing. IEEE/ACM Trans Netw 19(2):333–346. doi:10.1109/TNET.2010.2070844
17. Dubois-Ferriére H, Grossglauser M, Vetterli M (2011) Valuable detours: least-cost anypath routing. IEEE/ACM Trans Netw 19(2):333–346
18. Fuste-Vilella D, Garcia-Vidal J, Morillo-Pozo J (2008) Cooperative forwarding in IEEE 802.11-based MANETs. In: Wireless Days, 2008. WD ’08. 1st IFIP, pp 1–5. doi:10.1109/WD.2008.4812845
19. Hsu CJ, Liu HI, Seah WKG (2011) Opportunistic routing—a review and the challenges ahead. Comput Netw 55(15):3592–3603
20. Jain S, Das SR (2008) Exploiting path diversity in the link layer in wireless ad hoc networks. Ad Hoc Netw 6(5):805–825
21. Katti S, Rahul H, Hu W, Katabi D, Médard M, Crowcroft J (2006) 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, ACM, New York, pp 243–254
22. Katti S, Rahul H, Hu W, Katabi D, Médard M, Crowcroft J (2008) XORs in the air: practical wireless network coding. IEEE/ACM Trans Netw 16(3):497–510
23. Koutsonikolas D, Hu Y, Wang CC (2009) Pacifier: High-throughput, reliable multicast without “crying babies” in wireless mesh networks. In: INFOCOM 2009, IEEE, pp 2473–2481
24. Kurth M, Zubow A, Redlich JP (2008) Cooperative opportunistic routing using transmit diversity in wireless mesh networks. In: IEEE INFOCOM, pp 1310–1318. doi:10.1109/INFOCOM.2008.188
25. Larsson P (2001) Selection diversity forwarding in a multihop packet radio network with fading channel and capture. SIGMOBILE Mob Comput Commun Rev 5(4):47–54. http://doi.acm.org/10.1145/509506.509517
26. Larsson P (2001) Selection diversity forwarding in a multihop packet radio network with fading channel and capture. SIGMOBILE Mob Comput Commun Rev 5(4):47–54. doi:10.1145/509506.509517
27. Latouche G, Ramaswami V (1999) Introduction to matrix analytic methods in stochastic modeling. ASA-SIAM
28. Laufer R, Velloso P, Vieira L, Kleinrock L (2012) Plasma: a new routing paradigm for wireless multihop networks. In: INFOCOM, 2012 Proceedings IEEE, pp 2706–2710. doi:10.1109/INFCOM.2012.6195683

29. Le J, Lui J, Chiu DM (2008) Dcar: Distributed coding-aware routing in wireless networks. In: The 28th international conference on distributed computing systems, 2008. ICDCS '08. pp 462–469. doi:10.1109/ICDCS.2008.84
30. Le T, Liu Y (2010) Opportunistic overlay multicast in wireless networks. In: GLOBECOM, pp. 1–5
31. Li Y, Chen W, Zhang ZL (2009) Optimal forwarder list selection in opportunistic routing. In: IEEE 6th International Conference on mobile adhoc and sensor systems, 2009. MASS '09. pp 670–675. doi:10.1109/MOBHOC.2009.5336939
32. Li Y, Zhang ZL (2010) Random walks on digraphs: a theoretical framework for estimating transmission costs in wireless routing. In: 29th conference on information communications, INFOCOM'10, IEEE Press, pp 2775–2783
33. Liu H, Zhang B, Mouftah H, Shen X, Ma J (2009) Opportunistic routing for wireless ad hoc and sensor networks: Present and future directions. Commun Mag IEEE 47(12):103–109. doi: 10.1109/MCOM.2009.5350376
34. Lu M, Wu J (2009) Opportunistic routing algebra and its applications. In: IEEE INFOCOM, pp 2374–2382. doi:10.1109/INFCOM.2009.5062164
35. Luk CP, Lau WC, Yue OC (2008) An analysis of opportunistic routing in wireless mesh network. In: IEEE International Conference on Communications, 2008. ICC '08, pp 2877–2883. doi: 10.1109/ICC.2008.542
36. R Development Core Team: R: A language and environment for statistical computing. R foundation for statistical computing, Vienna, Austria (2008). http://www.R-project.org. ISBN 3-900051-07-0
37. Rozner E, Seshadri J, Mehta Y, Qiu L (2009) Soar: simple opportunistic adaptive routing protocol for wireless mesh networks. IEEE Trans Mob Comput 8(12):1622–1635. doi: 10.1109/TMC.2009.82
38. Shah R, Wietholter S, Wolisz A (2005) Modeling and analysis of opportunistic routing in low traffic scenarios. In: Third international symposium on modeling and optimization in mobile, ad hoc, and wireless networks, 2005. WIOPT 2005, IEEE, pp 294–304
39. Triviño-Cabrera A, Cañadas-Hurtado S (2011) Survey on opportunistic routing in multihop wireless networks. IJCNIS 3(2)
40. The Network Simulator ns-2. http://www.isi.edu/nsnam/ns (Last visited April 23, 2013)
41. WenZhong Y, ZhenYu Z, Bo W, XiaoHong W (2010) A reliable multicast for manets based on opportunistic routing. In: 2010 6th International Conference on wireless communications networking and mobile computing (WiCOM), pp 1–4
42. Westphal C (2006) Opportunistic routing in dynamic ad hoc networks: the oprah protocol. In: 2006 IEEE international conference on mobile adhoc and sensor systems (MASS), pp 570–573. doi:10.1109/MOBHOC.2006.278612
43. Wu J, Lu M, Li F (2008) Utility-based opportunistic routing in multi-hop wireless networks. In: The 28th international conference on distributed computing systems, 2008. ICDCS'08, pp 470–477
44. Wu Y, Das S, Chandra R (2007) Routing with a markovian metric to promote local mixing. In: INFOCOM 2007. 26th IEEE international conference on computer communications, IEEE, pp 2381–2385. doi:10.1109/INFCOM.2007.285
45. Yan Y, Zhang B, Mouftah H, Ma J (2008) Practical coding-aware mechanism for opportunistic routing in wireless mesh networks. In: IEEE international conference on communications, 2008. ICC '08, pp 2871–2876
46. Yan Y, Zhang B, Zheng J, Ma J (2010) Core: a coding-aware opportunistic routing mechanism for wireless mesh networks [accepted from open call]. Wirel Commun IEEE 17(3):96–103. doi:10.1109/MWC.2010.5490984
47. Yang S, Zhong F, Yeo CK, Lee BS, Boleng J (2009) Position based opportunistic routing for robust data delivery in manets. In: Global telecommunications conference, 2009. GLOBECOM 2009, IEEE, pp 1–6. doi:10.1109/GLOCOM.2009.5425351
48. Yanhua Li WC, Zhang ZL (2008) Design of forwarder list selection scheme in opportunistic routing protocol. Technical-report, University of Minnesota, Minneapolis, USA

49. Yuan Y, Yang H, Wong S, Lu S, Arbaugh W (2005) ROMER: Resilient opportunistic mesh routing for wireless mesh networks. In: IEEE workshop on wireless mesh networks (WiMesh)
50. Zeng K, Lou W, Yang J (2007) On throughput efficiency of geographic opportunistic routing in multihop wireless networks. Mob Netw Appl 12:347–357. doi:10.1007/s11036-008-0051-7
51. Zhao B, Seshadri RI, Valenti MC, Geraf HL, That HA (2004) Geographic random forwarding with hybrid-arq for ad hoc networks with rapid sleep cycles. In: Proceedings of IEEE Military Communication Conference (MILCOM)
52. Zhong Z, Wang J, Nelakuditi S, Lu GH (2006) On selection of candidates for opportunistic anypath forwarding. SIGMOBILE Mob Comput Commun Rev 10(4):1–2. doi:10.1145/1215976.1215978
53. Zorzi M, Rao R (2003) Geographic random forwarding (geraf) for ad hoc and sensor networks: multihop performance. IEEE Trans Mob Comput 2(4):337–348. doi:10.1109/TMC.2003.1255648

Chapter 12
Social-Based Routing Protocols in Opportunistic Networks

Ying Zhu and Yu Wang

Abstract Routing in opportunistic networks is very challenging as it must handle network partitions, long delays, and dynamic topology. Recently, the consideration of social characteristics of mobile nodes provides a new angle of view in the design of opportunistic routing protocols. In many opportunistic networks, a multitude of mobile devices are used and carried by people, whose behaviors are better described by social models. This opens the new possibilities of social-based routing, in which the knowledge of social characteristics of mobile users is used for making better opportunistic forwarding decision. In this book chapter, we briefly introduce several social properties (such as community, centrality, similarity, and friendship) which could be used for relay selection, and provide a broad survey of recent social-based opportunistic routing approaches.

12.1 Introduction

Opportunistic networks (Oppnets) [1–5] have recently drawn much attention from networking researchers due to the wide applications of these networks in challenging environments, such as intelligent transportation systems, mobile social networks, pocket switched networks, space communications, military operations, and mobile sensor networks. Intermittent connectivity in opportunistic networks results in the lack of instantaneous end-to-end paths, large transmission delay, and unstable network topology. These characteristics make the classical ad hoc routing protocols [6–8] not being applicable for opportunistic networks, since these protocols rely on

---

Y. Zhu · Y. Wang (✉)
Department of Computer Science, University of North Carolina, 9201 University City Blvd, Charlotte, NC 28223, USA
e-mail: yzhu17@uncc.edu

Y. Wang
e-mail: yu.wang@uncc.edu

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_12,

establishment of a complete end-to-end route from the source to the destination. On the other hand, wireless devices come into contact with each other opportunistically in opportunistic networks, which provides great opportunities to communicate wirelessly and exchange information.

Many new routing schemes [9–22] have been proposed for opportunistic networks or delay tolerant networks (DTNs). Most of them belong to three categories: *message-ferry-based, opportunity-based, and prediction-based.* In message-ferry-based methods [10–13], systems usually employ extra mobile nodes as ferries for message delivery. The trajectory of these ferries is controlled to improve delivery performance with store-and-carry. However, controlling these nodes leads to extra cost and overhead. In opportunity-based schemes [5, 14, 15], nodes forward messages randomly hop-by-hop with the expectation of eventual delivery, but with no guarantees. Generally, messages are exchanged only when two nodes meet at the same place, and multiple copies of the same message are flooded in the network to increase the chance of delivery. Predication-based routing protocols [16–22] make relay selection by estimating metrics relative to successful delivery, such as delivery probability or expected delay based on a history of observations. For more detail about opportunistic routing or predication-based DTN routing, please refer to other chapters in this book.

All of these opportunistic or DTN routing methods share a similar paradigm, the *store and forward* fashion. If there is no connection available at a particular time, a node can store and carry the data until it encounters other nodes. When the node has such a forwarding opportunity, all encountered nodes could be the candidates to relay the data. Thus, relaying selection and forwarding decision need to be made by the current node based on certain routing strategy. Various opportunistic or DTN routing approaches adopt different strategies based on various metrics most of which are estimated from the network's mobility pattern and encounter history. Examples of such metrics include estimated delivery probability to the destination node, expected delay, current network congestion level, or available network resources (such as bandwidth, storage, and energy). However, the unpredictable mobility and restricted resource in opportunistic networks significantly obstruct us from designing an ideal forwarding mechanism.

Lately, the consideration of social characteristics provides a new angle of view in the design of Oppnet routing protocols. In most Oppnet applications (mobile social networks [23–27] and pocket switched networks [28]), a multitude of mobile devices are used and carried by people, whose behaviors are better described by social models. This opens new possibilities of social-based routing for opportunistic networks, in which the knowledge of the social characteristics of mobile users are used to make better forwarding decisions. Note that social relations and behaviors among mobile users are usually long-term characteristics and less volatile than node mobility. Based on this observation and taking the recent advances in social network analysis, several social-based routing methods [29–35] have been proposed recently to exploit the various social characteristics in Oppnets or DTNs (such as community and centrality) to assist the relay selections.

In this chapter, we introduce the utilizations of social characteristics in the design of social-based routing protocols for opportunistic networks. The rest of the chapter is organized as follows. We first present several social properties, which can be extracted and used for opportunistic networks. Then, we review the current social-based approaches for routing in opportunistic networks. Finally, we conclude with a short summary and some discussions on the potential directions for future research.

12.2 Social Properties in Opportunistic Networks

We first introduce some social analysis methods and social properties related to social-based Oppnet/DTN routing. Many of these social properties have been studied in social network analysis.

12.2.1 Social Graph and Contact Graph

The most popular way to study the social relations among people and extract their social properties is building a *social graph* (also called social network). A social graph is a global mapping of everybody and how they are related. Such a graph is an abstract graph where vertices represent individual people and edges describe social ties between individual people. Social ties can be expressed in many forms. For example, different types of social ties may describe different social relationships among people such as friends, family members, and co-workers. Social graphs have been widely used in many applications, such as analysis of online social networks [36] or terrorist networks [37]. With a social graph, a variety of social metrics (e.g., communality, centrality, and similarity) can be easily calculated or estimated, and these metrics can be then used by social-based approaches. Therefore, it is crucial to obtain social graphs for social-based approaches.

A social graph is an intuitive source for many social metrics such as community and friendship. Unfortunately, it is not always available (due to either privacy or security reasons) or hard to be obtained via disclosed social data. However, with new networking technologies, we can study relationships among people by observing their interactions and behaviors over wireless networks. Building a *contact graph* is a common way to study the interactions among people in a network, and thus analyze their relationships and estimate the social metrics among them. In opportunistic networks, each possible packet forwarding happens when two mobile nodes are in contact (i.e., within transmission range of each other). By recording contacts seen in the past, a contact graph can be generated where each vertex denotes a mobile node (device or person who carries the device) and each edge represents one or more past meetings between two nodes. An edge in this contact graph conveys the information that two nodes encountered each other in the past. Thus the existence of an edge intends to have predictive capacity for future contacts. A contact graph can

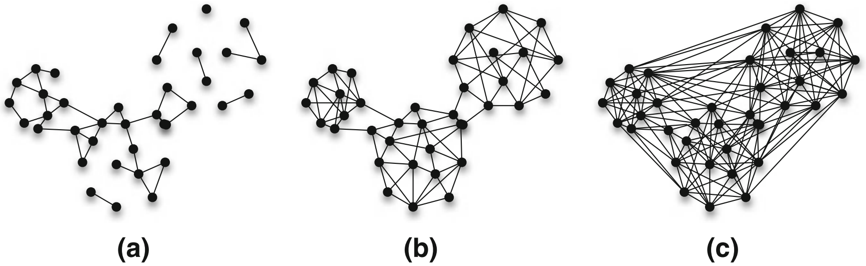
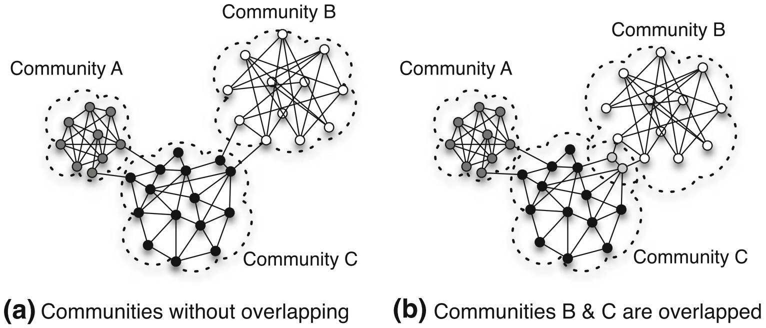


Fig. 12.1 Illustration of three different contact graphs constructed from the contact history in an opportunistic network using different aggregation periods

be constructed separately for each single time slot in the past, or it can be constructed to record the encounters in a specific period of time by assigning a set of parameters to each edge to record the time, the frequency, and the duration of these encounters. From the observation that people with close relationships such as friends, family members, etc., tend to meet more often, more regular, and with longer duration, we can extract nodes' relationships from the recorded contact graph for the opportunistic network, estimate their social metrics, and use such information to choose relays with higher probabilities of successful forwarding.

How to detect people's relationships and create the relative social graph from the recorded contact graph may affect estimation accuracy and the efficiency of social-based approaches. Most of the current social-based routing algorithms [28, 30, 31] directly treat the aggregated contact graph (merging the contact graphs of several time slots into one graph) as the social graph of all entities in the network, and uses this graph to generate social metrics for forwarding selection. This strategy is based on the observation that although the contact graph reflects the encounter history while the social graph reflects the social relations among people, the aggregated contact graph (the sum of contact graph over time), and the social graph are statistically similar. However, Hossmann et al. [38, 39] showed that the performance of these algorithms heavily depends on the way the graph is constructed out of observed contacts (i.e., contact aggregation) and proposed a method to select an appropriate aggregation period for contact aggregation. For example, in Fig. 12.1, three different contact graphs are generated from the same observed contact history with different aggregation durations. Figure 12.1a shows a disconnected contact graph which may be too sparse to detect any useful social structure, while Fig. 12.1c shows an almost complete graph which is too dense and thus useless. Figure 12.1b is a more appropriate aggregated contact graph, where three communities can be clearly detected. After building the aggregated contact graph, different social metrics can be obtained. For example, Hui et al. [40–43] proposed several community detection approaches (simple, k-clique, modularity, etc.) with great potential to detect both static and temporal communities. Bulut and Szymanski [34] introduced a method of detecting the quality of friendship by calculating the social pressure metric (SPM) from contact graphs.



(a) Communities without overlapping **(b)** Communities B & C are overlapped

Fig. 12.2 Illustration of community structures in contact/social graphs: **a** non-overlapping communities; **b** overlapping communities

12.2.2 Community

Community is an important concept in ecology and sociology [44–46]. In ecology, a community is an assemblage of two or more populations of different species occupying the same geographical area. In sociology, community is usually defined as a group of interacting people living in a common location. Community ecologists and sociologists study the interactions between species/people in communities at many spatial and temporal scales [44–48]. It has been shown that a member of a given community is more likely to interact with another member of the same community than with a randomly chosen member of the population [48]. Therefore, communities naturally reflect social relationship among people. Figure 12.2 illustrates examples of two types of community structures in social graphs.

Since wireless devices are usually carried by people, it is natural to extend the concept of social community into opportunistic networks to explore interactions among wireless devices. It is believed that devices within the same community have higher chances to encounter each other. Therefore, the knowledge of community structures could help a routing protocol to choose better forwarding relays for particular destinations, and hence improve the chance of delivery. Many proposed community detection algorithms [40–43, 49–55] are available for identifying social communities from the contact graph of opportunistic networks. Some common detection methods for communities are summarized as follows.

Minimum-cut method [49]: The minimum-cut method divides the social graph or the network into a predetermined number of components such that the number of edges between components is minimized.

Hierarchical clustering: The hierarchical clustering method uses the concept of similarity to detect communities. The similarity metric aims to measure the degree of similarity (usually topological) between node pairs. Common calculation methods include the cosine similarity, the Jaccard similarity coefficient, and the Hamming distance between rows of the adjacency matrix of the network. A common community detection strategy is: all nodes within a community have similarity greater than a given threshold.

Girvan-Newman algorithm [50]: Girvan and Newman proposed a community detection method using a graph-theoretic metric, *betweenness* (we will introduce later), to identify the bridge edges among communities in the network. By removing these bridge edges, the communities can be easily detected.

Modularity maximization [47, 51, 52]: The modularity maximization method detects communities by searching over all possible network divisions to find the one with particularly high modularity. Modularity is defined as a benefit function, which measures the quality of a particular division. Optimization methods (such as greedy algorithms or simulated annealing) are often used because exhaustive search over all possible division is usually too expensive.

The Louvain method [53]: The Louvain method is a greedy optimization method with two phases. It first looks for small communities by locally optimizing modularity, it then aggregates nodes in the same community, and builds a new network whose nodes are the communities. These two steps are repeated iteratively until a maximum modularity is achieved.

Clique-based methods [54]: In a graph, a clique is a subgraph in which every node is connected to every other node in the subgraph. Since a clique is the most tightly connected structure, there are many community detection approaches based on the detection of cliques in a graph. As a node can belong to multiple cliques, these methods may lead to overlapping community structures.

12.2.3 Centrality

In graph theory and network analysis, *centrality* is a quantitative measure of the topological importance of a vertex within the graph. A central node, typically, has a stronger capability of connecting other nodes in the graph. In a social graph, the centrality of a node describes the social importance of its represented person in the social network. In opportunistic networks, the sociological centrality metrics [56] can also be used for relay selections (nodes with high centralities are always good candidates of relay nodes).

There are several ways to define centrality in a graph. Three common centrality measures are *degree centrality*, *betweenness centrality*, *and closeness centrality* [57–59]. *Degree centrality* is the simplest centrality measure which is defined as the number of links (i.e., direct contacts) incident upon a given node. For example, in Fig. 12.3a, the degree centralities of node a and node b are 3 and 4 respectively while those of the other nodes are 1. A node with a high degree centrality is a popular node with a large number of possible contacts, and thus it is a good candidate of a message forwarder for others (i.e., a hub for information exchange among its neighborhood). *Betweenness centrality* measures the number of shortest paths passing via certain given node. It is a more useful measure of the load placed on the given node in a network as well as the node's importance to the network than just connectivity.

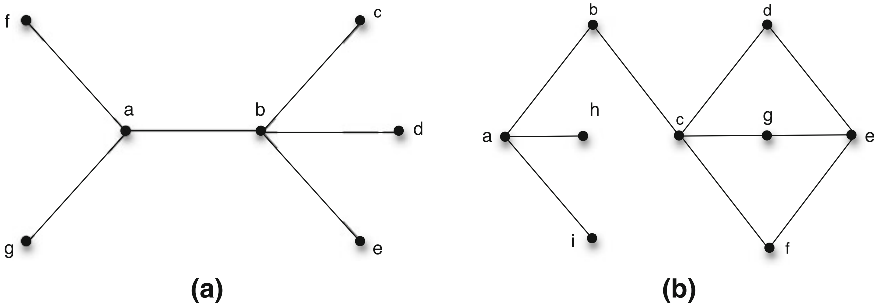


Fig. 12.3 Illustration of centrality and similarity measurements over simple social graphs: **a** three centrality measures; **b** similarity measurement

For example, the betweenness centralities of a and b in Fig. 12.3a are 18 and 24 (9 and 12 if do not consider the direction of paths), respectively. But for the other nodes, their betweenness centralities are 0 since they are not on any shortest paths. The betweenness centrality can also be normalized between 0 and 1 by dividing the number of all possible paths. Nodes that occur on many shortest paths between other nodes have higher betweenness than those that do not. A node with high betweenness centrality can control or facilitate many connections between other nodes, thus it is ideal for a bridge node during message exchange. The *closeness centrality* of a node is defined as an inverse of its average shortest distance to all other nodes in the graph. If a node is near to the center of the graph, it has higher closeness centrality and is good for quickly spreading messages over the network. For the example in Fig. 12.3a, the closeness centrality of node a is $2/3$ since its average shortest distance to all others is 1.5. The closeness centralities of b, $c/d/e$, and f/g are $3/4$, $6/13$, and $3/7$, respectively.

12.2.4 Similarity

Similarity [30] is a measurement of the degree of separation. It can be measured as the number of common neighbors between individuals in social networks. For example, in Fig. 12.3b, the similarity between a and c is 1, while that between c and e is 3. Sociologists have long known that there is a higher probability of two people being acquainted if they have one or more other acquaintances in common. In a network, the probability of two nodes being connected by a link is higher when they have a common neighbor. When the neighbors of nodes are unlikely to be in contact with each other, diffusion can be expected to take a longer time. In addition, there are other ways to define the similarity beyond common neighbors, such as similarity on user interests [60] and similarity on user locations [61].

12.2.5 *Friendship*

Friendship is another concept in sociology, which describes close personal relationships. In opportunistic networks, friendship can be defined between a pair of nodes. On the one hand, to be considered as friends of each other, two nodes need to have long-lasting and regular contacts. On the other hand, friends usually share more common interests as in the real world. In sociology, it has been shown that individuals often befriend others who have similar interests, perform similar actions, and frequently meet with each other [62]. This observation is called *homophily phenomenon*. Therefore, the friendship in opportunistic networks can be roughly determined by using either contact history between two nodes [34] or common interests/contents claimed by two nodes [33].

12.3 Social-Based Routing for Opportunistic Networks

In this section, we introduce several social-based routing methods for opportunistic networks that take advantage of less volatile social characteristics.

12.3.1 *Label Routing*

Hui and Crowcroft [29] introduced a routing method (called as label routing hereafter) based on community labels in Pocket Switched Networks (PSNs). A PSN [28] is a type of opportunistic networks where mobile devices are carried by people and communicate with each other when people meet. To reduce the amount of traffic created by forwarding messages in PSNs, the proposed routing method uses a labeling strategy to select forwarding relay. Since people in the same community are likely to meet regularly, they are appropriate forwarders for messages destined to the members of their community. In their solution, Hui and Crowcroft assumed that each node has a small label telling others about its affiliation/group (i.e., its social community), just like name badges used in a conference. Based on the labels, label routing chooses to forward messages to destinations directly or to next-hop nodes, which belong to the same group (label) with the destinations.

Label routing takes the advantage of the knowledge of social community. It assumes that people from the same community tend to meet more often than people from different communities, and hence can be good forwarders to relay messages destined to the other members in the same community (with the same label). Label routing requires very little information about each individual (only its group/affiliation). This is easy to implement in PSN applications, by tapping a mobile device and writing down the affiliation of the owner. In other words, the community (or group) information relies on user inputs in label routing. However, user-defined communities may not always reflect the position/contact relationship among nodes. For example, two nodes in the same community may be physically far away and could never meet with each other. In this scenario, using one node to be the forwarder for the other may not be a good choice. In addition, in label routing, the message forwarding from the source to the destination is purely via the members within the same community of the destination. This may significantly increase the delay or even fail to deliver the message. For instance, message delivery will fail when the source does not meet any member from the destination's community, even though there are possible relay nodes from other communities. Recently, there are new social-based routing methods using multiple social labels [63] or multi-level social group information [64] to enhance the chances of delivery.

12.3.2 SimBet Routing

Daly and Haahr [30] proposed a social-based routing protocol (called SimBet routing hereafter) which uses *betweenness centrality* and *similarity* metrics to identify some "bridge" nodes (with high values of these metrics) in networks. To avoid exchanging information about the entire network topology, they only estimated the betweenness centrality Bet_n for each node n in its local neighborhood. For similarity metric, they considered the similarity $Sim_n(d)$, the number of common neighbors, of the current node n with the destination node d. Both the social metrics are maintained and updated dynamically in opportunistic networks. Therefore, the proposed SimBet routing makes forwarding decision by considering not only the pre-estimated betweenness centrality metric but also the locally determined social similarity. Nodes with high betweenness centralities are those nodes that can act as bridges in their neighborhood, while nodes with high similarities with the destination are more likely to find a common neighbor with the destination that can act as the forwarder.

In SimBet routing, when a node n meets another node m and holds a message with destination d, n calculates its relative betweenness utility and similarity utility to node m as follows:

$$SimUtil_n = \frac{Sim_n(d)}{Sim_n(d) + Sim_m(d)}; \quad BetUtil_n = \frac{Bet_n}{Bet_n + Bet_m}. \tag{12.1}$$

Then node n can compute its SimBet utility, which is a weighted combination of betweenness utility and similarity utility:

$$SimBetUtil_n(d) = \alpha SimUtil_n(d) + (1 - \alpha) BetUtil_n. \tag{12.2}$$

Here, α is a tunable parameter that can adjust the relative importance of the two utilities. For the message with d as its destination, if $SimBetUtil_m(d) > SimBetUtil_n(d)$, node n forwards the message to node m. Otherwise, it contin-

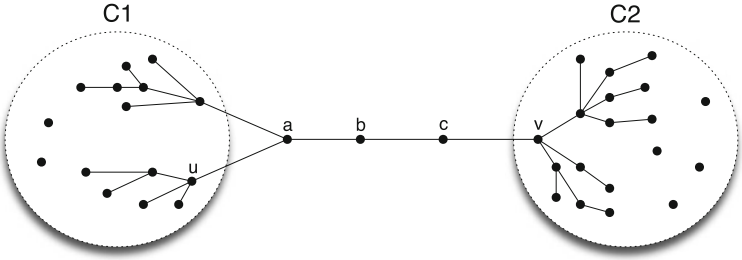


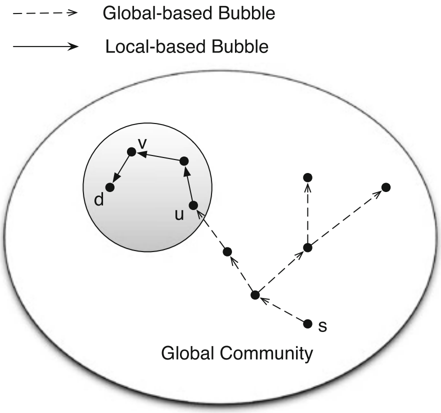
Fig. 12.4 Illustration of problems of the SimBet routing

ues to hold the message. Via possible multi-hop relays, the message may eventually reach d.

In summary, SimBet routing uses two social metrics (centrality and similarity) to estimate or predict the probability that potential relay nodes may meet the destination. It is obvious that both metrics are effective at identifying suitable relays in different scenarios respectively. Take an example graph, as shown in Fig. 12.4, where a few low-degree bridges (i.e., a, b, and c) connect two well-connected components C_1 and C_2. Assume that node u wants to send a message to node v. When node u encounters node a, it compares its SimBet utility with that of node a's. Both u and a have zero similarity to v, but u's global betweenness centrality is less than a's since a sits on more of the shortest paths. Thus, u will transfer the message to a based on SimBet routing. In this case, centrality metric helps to pick the better relay node. On the contrary, if node a wants to send a message to v and it encounters node b, similarity metric will play a role since the global betweenness centralities of a and b are the same. Therefore, a has a smaller similarity (zero common neighbor) to v than b has (one common neighbor with v). Therefore, combining multiple social metrics may make the social-based protocol more effective in broad situations. However, due to the uncertainty of future encounters and underlying social graph, it is still possible that the node with high SimBet utility fails to deliver the message to the destination.

To avoid global information exchanges, SimBet routing provides a distributed method to calculate social metrics locally, which is desirable in a mobile environment. However, estimating centrality-based solely on local information may lead to inaccurate "bridge" identification. For instance, in the example shown in Fig. 12.4, it is assumed that u wants to send a message to v. When u encounters node a, based on the two-hop information, u's local betweenness Bet_u is much larger than a's Bet_a. Since both u and a have zero similarities to v, the overall $SimBetUtil_u(v) > SimBetUtil_a(v)$. Then, node u will not pass this message to node a, and thus miss the opportunity to deliver the message. Nonetheless, considering global betweenness, each of the nodes of a, b, and c has highest betweenness in the entire network (since they form the only path connecting components C_1 and C_2), and can then be correctly identified. A possible way to increase the chance of correct "bridge" identification is using larger neighborhood information, although this may increase communication cost. Similarly, to increase the chance of delivery, multiple

Fig. 12.5 An illustration of the Bubble Rap Forwarding from source s to destination



relay nodes could be used. The trade-off is always between delivery performance and communication cost.

12.3.3 Bubble Rap Forwarding

The forwarding strategy, *Bubble Rap Forwarding*, proposed by Hui et al. [31], also relied on two social characteristics (community and centrality). They assumed that each node belongs to at least one community and its node centrality (either betweenness or degree centrality) in the community describes the popularity of the node within this community. Each node has a global centrality across the whole network (or called global community), and a local centrality within its local community. A node may also belong to multiple communities and hence have multiple local centralities. Taking advantages of these social characteristics, Bubble Rap Forwarding basically includes two phases: a bubble-up phase based on global centrality and a bubble-up phase based on local centrality. In both phases, the bubble-up forwarding strategy is utilized to forward messages to nodes which are more popular than the current node (i.e., with higher centrality). When a node s has a message with destination of d, it first bubbles the message up based on the global centrality, until the message reaches a node which is in the same local community C_d as the destination d. This procedure is shown as dashed arrows in Fig. 12.5. After the message reaches d's community at node u, Bubble Rap Forwarding switches to the second phase which uses members of C_d as relays. This forwarding strategy continues to bubble up the message through the local community based on local centrality until the destination is reached. This later procedure is shown as solid arrows in Fig. 12.5. In order to reduce cost, it is also required that whenever a message is delivered to the community, the original carrier delete this message from its buffer to prevent further dissemination.

Bubble Rap Forwarding uses the concept of community in addition to node centrality to help with the forwarding decision. The introduction of local centrality inside a community is more beneficial than local centrality around local neighborhood (i.e., k-hop) [30]. The bubble-up operations allow fast transfer of a message toward the destination or its community. However, such a strategy may fail when the destination belongs only to the communities whose members are all with low global centrality values. In this case, the bubble-up process in the first phase of Bubble Rap Forwarding cannot find the relay node which is in the same local community as the destination node. A possible solution for this problem to have a timeout timer for bubble-up process and exchange to other backup strategy for data delivery after timeout.

In addition, it is interesting to study how to handle hierarchical communities where the destination d may belong to multiple overlapping communities. In that scenario, when the current encountering node u shares multiple communities with d, one of d's local communities need to be picked to bubble-up. A simple solution is picking the local community with which d has the highest centrality.

12.3.4 Social-Based Multicasting

All of the social-based routing methods discussed above are unicast routing protocols. Social-based approaches can be applied to multicast routing protocols for Oppnets/DTNs as well. Recently, Gao et al. have [32] proposed a set of multicast routing methods, which use both centrality metric and community for relay selection.

Instead of using traditional centrality metrics such as betweenness, Gao et al. introduced a new metric (called *cumulative contact probability*) based on the Poisson modeling of social networks. Betweenness is purely defined based on the topology of the contact graph, and may not be sufficient to represent the probabilities for a node to contact others. Thus, a weighted social network model is used to differentiate the contact frequencies of different node pairs. In such model, the contact process of each node pair (i, j) is formulated as a Possion process with an average contact rate of $\lambda_{i,j}$. Then the cumulative contact probability of node i can be defined as $C_i = 1 - \frac{1}{N-1} \sum_{j=1, j \neq i}^{N} e^{-\lambda_{ij} T}$. Here, N is the total number of nodes in the network, and T is the total time period. This centrality metric (or its variation) is used in the proposed multicast methods to select a relay node with higher centrality.

In [32], two multicast problems are considered: single-data multicast and multiple-data multicast, whose goals are to deliver a data item or a set of data items to a set of destinations within the time constraint T. The additional optimization objective is to minimize the number of relays used to achieve the average delivery ratio p.

For single-data multicast, the authors assumed the destinations are uniformly distributed, and thus they tried to ensure that all nodes are contacted by the data source or its selected relays within T. Based on cumulative contact probabilities (i.e., centrality) of nodes, a minimal number of relay nodes are selected among the contacted neighbors of data source to guarantee the average delivery ratio is larger than p. This selection problem can be solved as a unified knapsack problem. The

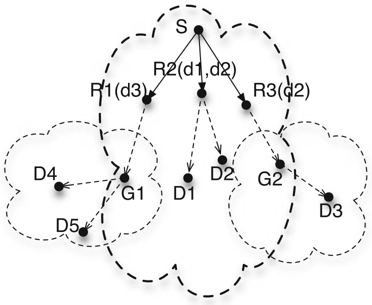


Fig. 12.6 MDM data forwarding process (Relays $R1$, $R2$, and $R3$ are selected by S)

centrality metric is also refined for the case where the data source does not meet its contacted neighbors.

For multi-data multicast, the authors proposed a community-based approach which only requires nodes to maintain the probabilities of forwarding each data item to other nodes in the same community. When the destinations are in other communities, data forwarding is conduced through some gateway nodes which belong to multiple communities. The data source selects relays among its contacted neighbors based on the local centrality metric, and places appropriate data items on each relay. They used a two-stage relay selection scheme, where both data item selection and relay selection are modeled as knapsack problems. Their method can ensure that the average delivery probability is larger than p. Figure 12.6 illustrates such a process. Source node S needs to multicast three data items $d1$,$d2$,$d3$ to destination sets $\{D1\}$,$\{D2, D3\}$, $\{D4, D5\}$ respectively. The relay and data item selections are shown in the figure. Destinations $D1$ and $D2$ are in the same community with relay $R2$, so $R2$ forwards $d1$ and $d2$ to the destinations according to its local destination-awareness about $D1$ and $D2$. For destination $D3$, $D4$, and $D5$, because they belong to other communities, data forwarding goes through the "gateway" nodes $G1$ and $G2$, which belong to multiple communities.

In summary, the solutions proposed in [32] provide new multicast strategies for Oppnet/DTNs based on the new centrality metric and community concept. The cumulative contact probability based on the Poisson modeling of social networks represents more accurate measurement of the probabilities for a node to contact other nodes. But this new model increases computational complexity as well. The authors addressed the optimization problem to minimize the number of relay nodes while satisfying the required delivery ratio and time constraint.

12.3.5 Homophily Based Data Diffusion

Zhang and Zhao [33] proposed a data diffusion scheme based on the "homophily" phenomenon in social networks. Here, data diffusion aims to deliver data to all nodes in the network, data may not be completely delivered from one node to another during a contact between them, since the contact time is too short to transmit the data or the buffer available at the receiving node is insufficient to hold the data. Therefore, in the design of data diffusion protocol, not only the contact probability between nodes but also the data propagation orders (which data should be propagated first) affect the diffusion speed and data access delay.

To choose an appropriate relay node to diffuse and an appropriate data item to buffer, Zhang et al. introduced a method using the *friendship* among nodes and the "homophily" phenomenon. The "homophily" phenomenon describes the trend in the real word that friends usually share more common interests than strangers. By applying the same idea from "homophily" phenomenon, their proposed data diffusion strategy diffuses the most similar data items between friends, and diffuses the most different data items between strangers. If a node meets a new contact who is a friend, it first diffuses the most similar data items of their common interests to its friend first until the contact time is over. If the new contact is a stranger, it starts from the data item most different from their common interest. By theoretical analysis, Zhang et al. showed that this data diffusion scheme achieves better diffusion speed and data access delay than the other three possible schemes (including diffusing the most similar data to any encounter, diffusing the most different data to any encounter, and diffusing the most different data between friends and the most similar data between strangers).

This proposed method provides a new angle to social-based approaches. It considers the need for managing data propagation orders which is an important aspect of design issues in opportunistic networks. With the same amount of communication opportunity and duration, more useful information can be transmitted under this proposed method. In addition, the proposed method is not conflicted with other Oppnet/DTN routing protocols. It can be used together with other Oppnet/DTN routing protocols to make better relay decisions with efficient data propagation orders. In the proposed method, social friendship is the only metric used to predict the encounter's needs of information and friendship is user defined. However, user defined friendship is not always available. Therefore, it is another challenging direction need to be further explored regarding how to efficiently detect the friendship in dynamic Oppnets.

12.3.6 Friendship-Based Routing

Bulut and Szymanski [34] also used friendship to aid the delivery of packets in DTNs. They introduced a new metric, *social pressures metric* (SPM), to accurately detect the quality of friendship. Different from [33], where friendship is defined by users based on their social relationships, this approach considered friends as nodes which contact each other frequently and have long-lasting and regular contacts. Therefore, the social pressures metric between nodes i and j can be estimated from the encounter histories of these nodes (recorded by the nodes) as:

$$SPM_{i,j} = \frac{\int_{t=0}^{T} f(t)dt}{T}, \tag{12.3}$$

where $f(t)$ denotes the remaining time to the first encounter of these nodes after time t and T is the total time period. SPM describes the average forwarding delay if node i has a message destined to j at each time unit. Then, the link quality $w_{i,j}$ between each pair of nodes, (i, j), is defined as $w_{i,j} = \frac{1}{SPM_{i,j}}$. The authors assumed that the bigger value of $w_{i,j}$ represents the closer friendship between i and j. Using the value of $w_{i,j}$, each node can construct its friendship community for each period T as a set of nodes whose link quality with itself is larger than a threshold. When a node i, having a message destined to d, meets with node j, it forwards the message to j if and only if (1) j and d are in the same friendship community (in the current period) and (2) j is a stronger friend of d than i.

In summary, this friendship-based routing method uses the node contact information in each period to calculate the friendship metric (i.e., SPM), and constructs the friendship community. These social metrics can indeed help with making smarter forwarding decisions. However, the calculation of these metrics needs the whole contact information during each period, which may not be realistic in most DTNs/Oppnets. To obtain $f(t)$ in the current period, node i needs to know the time of its first encounter to node j after time t in this period, which is an event in future. Therefore, either the values in contact history from previous periods are used for this calculation at the current period or the estimated future contacts in this period are available for this calculation. This is clearly a drawback of this proposed method. In addition, this friendship-based routing uses a similar forwarding scheme to label routing [29], which may lead to the same problem. If the source node fails to meet with any node in the same friendship community with the destination node, the delivery fails. Therefore, more felicitous forwarding strategies should be studied for this friendship-based routing.

Although the friendship-based method [34] and homophily-based method [33] both use friendship metrics for delivery data in Oppnet/DTNs, they are designed for different purposes. In [33] the friendship measurement is used to select which data items to diffuse, while in [34] the friendship metric is used to detect communities and select which relay nodes to forward. Therefore, different social metrics or various calculation methods need to be designed for specific design purposes. There is no universal solution for all applications.

12.3.7 Social-Aware and Stateless Routing

In [60] Mei et al. took advantage of the observation that people with similar interests tend to meet more often, to propose a *social-aware and stateless routing* (SANE) for pocket switched networks. This routing strategy characterizes an individual u by its interest profile, which is a k-dimensional vector I_u. To express the interest similarity between two individuals u and v, the *cosine similarity* is defined as, $\Theta(I_u, I_v) = \cos(\angle I_u I_v) = \frac{I_u I_v}{\|I_u\| \|I_v\|}$. It quantitatively measures "homophily", the degree of interest similarity, with higher value of $\Theta(I_u, I_v)$ corresponding to higher "homophily" degree.

In SANE, a message should be forwarded to individuals whose interest profiles closely resemble that of the destination.They assume that the interest profile of a message m is the interest profile of its destination; the node is allowed to forward to other relays no more than N times. Thus, a message m will be relayed to a node u only if the *cosine similarity* of the interest profile between message m and node u is higher than a given threshold ρ and the message has been forwarded to no more than N nodes.

One of the advantages of this method is that each node only needs to maintain the interest profile without extra storage. The cost of maintaining and updating this social metric is also relevantly easy. These advantages improve the scalability of this routing method. The cost of the message delivery C_m can be controlled by the setting of threshold ρ and the number of copies N. C_m will increase proportional to N. And as the threshold ρ decreases, both the delivery success probability and C_m will increase, meanwhile the delivery delay will decrease.

12.3.8 User-Centric Data Dissemination

Gao and Cao [65] proposed a user-centric data dissemination approach which aims to maximizes the cost-effectiveness while satisfying the user interests. In order to select relays which have better capability of disseminating data to interesters, they create their own concept of *centrality value*. Different from the concept of centrality used in [30–32], *centrality value* considers both social contact patterns and user interests simultaneously.

The *interest profile* of a node i is defined as a $M \times 1$ probability vector $P_i = [p_{i1}, \ldots, p_{iM}]^T$, where p_{ij} indicates the user probability to be interested in the j-th *keyword*. Similarly, a *data item* is also described by M *keywords* $k_1, \ldots, k_n \in K$ and *weights* w_{k_i} indicates the importance of *keyword* k_i in describing the data. Thus the intertest probability of node i in data D is $p_i = P_i^T D = \sum_{j=1}^{M} w_{k_i} p_{i,k_j}$. The *centrality value* of a node i for the *data item* d_k at time $t \ll T_k$ is defined as

$$C_i^{(k)}(t) = \sum_{j \in N_i} p_j^{(k)} C_{ij}(T_k - t). \tag{12.4}$$

Here, T_k is the time constraint of data d_k, t is the current time, N_i is the set of nodes whose information is maintained by node i, and $C_{ij}(T_k - t)$ is the probability that node i can forward data d_k to node j within time $T_k - t$. $C_i^{(k)}(t)$ indicates the expected number of interesters (nodes interested in the data item d_k) that node i can encounter during the remaining time $T_k - t$ of data dissemination. Here, node i maintains

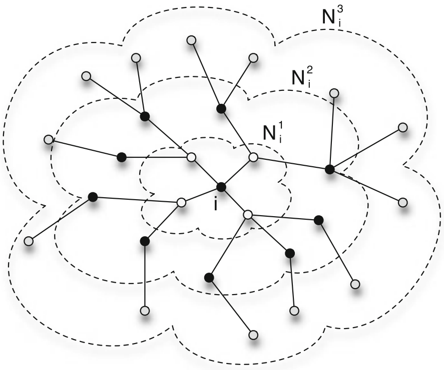


Fig. 12.7 Scope of network information maintained by node *i*

the network information including the contact patterns and interest profiles of all the nodes in N_i. The calculation of $C_{ij}(T_k - t)$ depends on the scope of information maintained by i. Figure 12.7 illustrates various scopes of network information, where N_i^r is the r-hop neighborhood of node i on the contact graph. If the calculation of $C_{ij}(T_k - t)$ exploits N_i^1, only the local network knowledge is considered and the *local centrality values* are obtained. If the calculation of $C_{ij}(T_k - t)$ exploits N_i^r, the r-hop centrality values are obtained.

After getting the *centrality values*, their relay selection makes sure that a new relay always has better capability of disseminating data to interesters than the existing relays based on this newly defined time-varying centrality. A node i is selected as the relay by another relay j for data d_k at time t only if selecting i increases the cost-effectiveness ratio $\frac{N_I^k(t)}{N_R^k(t)}$ estimated at j. That is, $\frac{N_I^k(t)+C_i^{(k)}(t)}{N_R^k(t)+1} \gg \frac{N_I^k(t)}{N_R^k(t)}$, which is equal to $C_i^{(k)}(t) \gg \frac{N_I^k(t)}{N_R^k(t)}$. Here $N_R^k(t)$ is the number of selected relays for data d_k at time t, $N_I^k(t)$ is the estimation at time t on the number of interesters that will receive d_k by time T_k.

This strategy considers social contact patterns with user interests. With the consideration of multiple elements, it may lead to more accurate estimation of forwarding probability.

12.3.9 Sociability-Based Routing

In [66], Fabbri and Verdone proposed a sociability-based DTN routing, which is based on the idea that nodes with high degrees of sociability (frequently encountering many different nodes) are good forwarding candidates. They defined the *sociability indicator* metric to evaluate the forwarding ability of a node. This metric quantifies the social behavior of a node by counting its encounters with all the other nodes in the network over a period T and is therefore a time-varying parameter. The routing

Table 12.1 Comparisons of social-based routing protocols for opportunistic networks

Routing protocols	Community	Centrality	Similarity	Friendship	Interest
Label [29]	√				
SimBet [30]		√	√		
Bubble rap [31]	√	√			
Social-based muticasting [32]	√	√			
Homophily-based data diffusion [33]				√	√
Friendship-based routing [34]	√			√	
SANE [60]			√		√
User-centric data dissemination [65]		√			√
Sociability-based routing [66]		√			

strategy forwards packets to the most sociable nodes only. It is worth to notice that the strategy also considers both *first hop-based sociability* and *k-th hop-based sociability*. For k-th hop-based sociability, the highly sociable neighbors are considered during the calculation of a user's sociability.

12.4 Summary

In this chapter, we introduce different social characteristics of mobile nodes in opportunistic networks and current social-based routing protocols which use these social characteristics to assist packet forwarding. Table 12.1 summarizes the social characteristics used by these routing protocols. From the analysis and comparison of these methods, we can conclude that social-based approaches are more promising than pure opportunity-based routing protocols for opportunistic networks since these social-based approaches take advantages of relatively stable characteristics (social properties) efficiently to predict and deal with the dynamics of opportunistic networks. Nonetheless, none of these approaches can guarantee the perfect routing performance (such as delivery guarantee or minimum delay). The prediction always has certain probability to fail in some cases. Besides, the performance of a routing protocol in opportunistic networks depends heavily on the mobility model, environment, node density, social structure, and many other facts. Therefore, a universal routing solution for all Oppnet application scenarios is extremely hard. In other words, for certain Oppnet applications, we need to design specific routing protocols and mobility/social models to fulfill the requirements.

Although social-based routing in opportunistic networks has lately received much attention in the wireless network community as a relatively new area, there are still quite a few challenges left. For instance:

- *How to explore new social characteristics?* It is interesting to see whether there are other more realistic and accurate social characteristics beyond community, centrality, and friendship, which can be used to further improve routing performance in Oppnets (even if it would be more complex).

- *How to use multiple social characteristics efficiently?* Most of the current social-based approaches use only simple definitions of one or two social characteristics (such as k-clique for community, node degree for centrality, contact frequency for friendship). Is it possible to improve the Oppnet routing performance by using more social characteristics together? Is there any trade-off?
- *How to model and extract accurate social characteristics in dynamic opportunistic networks?* Social-based approaches significantly rely on accurate modeling of the social characteristics used by them. However, due to the lack of continuous connectivity and time-varying topology, it is hard to accurately estimate certain social characteristics without global or future information even with simple definitions.
- *What if the social relationships do not reflect the real contact opportunity?* Although in most cases strong social relation indicates high contact probability, there are some exceptions. How to figure out such exceptions to avoid wrong routing decisions in social-based approaches? How to combine social-based approaches with other types of opportunistic routing or DTN routing to achieve better performance over wider scenarios?

Recently, researchers have tended to improve Oppnet routing by exploring stateless social characteristics [60], combining multiple social characteristics and using other information as the complement of social-based metrics. Finally, we also believe that social-based approaches will be applied to wider research topics in communication networks far beyond routing protocols in opportunistic networks.

Acknowledgments This work is supported in part by the US National Science Foundation under Grant No. CNS-0915331 and CNS-1050398.

References

1. Conti M, Kumar M (2010) Opportunities in opportunistic computing. IEEE Comput 43(1): 42–50
2. Pelusi L, Passarella A, Conti M (2006) Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. IEEE Commun Mag 44(11):134–141
3. Zhang Z (2006) Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges. IEEE Commun Surv Tutor 8(1):24–37
4. Wang Y, Dang H, Wu H (2007) A survey on analytic studies of delay-tolerant mobile sensor networks. Wirel Commun Mob Comput 7(10):1197–1208
5. Juang P, Oki H, Wang Y, Martonosi M, Peh L-S, Rubenstein D (2002) Energy-efficient computing for wildlife tracking:design tradeoffs and early experiences with Zebranet. SIGOPS Oper Syst Rev 36(5):96–107
6. Perkins CE, Royer EM (1999) Ad-hoc on-demand distance vector routing. In: Proceedings of the 2nd IEEE workshop on mobile computing systems and applications, 1999
7. Johnson DB, Maltz DA (1996) Dynamic source routing in ad hoc wireless networks. In: Imielinski T, Korth HF (eds) Mobile Computing. Kluwer Academic Publishers, Norwood, pp 153–181
8. Royer EM, Toh C-K (1999) A review of current routing protocols for ad-hoc mobile wireless networks. IEEE Pers Commun 6(2):46–55

9. Yuan Q, Cardei I, Wu J (2009) Predict and relay: an efficient routing in disruption-tolerant networks. In: Proceedings of the 10th ACM international symposium on mobile ad hoc networking and computing (MobiHoc '09), pp 95–104, 2009
10. Shah RC, Roy S, Jain S, Brunette W (2003) Data mules: modeling a three-tier architecture for sparse sensor networks. In: Proceedings of the 1st IEEE international workshop on sensor network protocols and applications, pp 30–41, 2003
11. Burns B, Brock O, Levine BN (2005) MV routing and capacity building in disruption tolerant networks. In: Proceedings of the 24th IEEE international conference on computer communications (INFOCOM), pp 398–408, 2005
12. Zhao W, Ammar M, Zegura E (2004) A message ferrying approach for data delivery in sparse mobile ad hoc networks. In: Proceedings of the 5th ACM international symposium on mobile ad hoc networking and computing (MobiHoc '04), pp. 187–198, 2004
13. Zhao W, Ammar M, Zegura E (2005) Controlling the mobility of multiple data transport ferries in a delay-tolerant network. In: Proceedings of the 24th IEEE international conference on computer communications (INFOCOM), 2005
14. Vahdat A, Becker D (2006) Epidemic routing for partially connected ad hoc networks. Duke University Technical report CS-200006
15. Spyropoulos T, Psounis K, Raghavendra CS (2005) 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), pp 252–259, 2005
16. Lindgren A, Doria A, Schelén O (2003) Probabilistic routing in intermittently connected networks. SIGMOBILE Mob Comput Commun Rev 7(3):19–20
17. Zhang X, Neglia G, Kurose J, Towsley D (2007) Performance modeling of epidemic routing. Comput Netw 51(10):2867–2891
18. Burgess J, Gallagher B, Jensen D, Levine BN (2006) Maxprop: routing for vehicle-based disruption-tolerant networks. In: Proceedings of the 25th IEEE international conference on computer communications (INFOCOM), 2006
19. Leguay J, Friedman T, Conan V (2006) Evaluating mobility pattern space routing for DTNs. In: Proceedings of the 25th IEEE international conference on computer communications (INFOCOM), Barcelona, Spain, 2006
20. Liu C, Wu J (2008) Routing in a cyclic mobispace. In: Proceedings of the 9th ACM international symposium on mobile ad hoc networking and computing (MobiHoc '08), pp 351–360, 2008
21. Liu C, Wu J (2007) Scalable routing in delay tolerant networks. In: Proceedings of the 8th ACM international symposium on mobile ad hoc networking and computing (MobiHoc '07), pp 51–60, 2007
22. Liu C, Wu J (2009) An optimal probabilistic forwarding protocol in delay tolerant networks. In: Proceedings of the 10th ACM international symposium on mobile ad hoc networking and computing (MobiHoc '09), pp 105–114, 2009
23. Chaintreau A, Fraigniaud P, Lebhar E (2008) Opportunistic spatial gossip over mobile social networks. In: Proceedings of the 1st ACM workshop on online social networks (WOSN '08), pp 73–78, 2008
24. Chierichetti F, Lattanzi S, Panconesi A (2008) Gossiping (via mobile?) in social networks. In: Proceedings of the 5th ACM international workshop on foundations of mobile computing (DIAL M-POMC '08), pp 27–28, 2008
25. Miluzzo E, Lane ND, Fodor K, Peterson R, Lu H, Musolesi M, Eisenman SB, Zheng X, Campbell AT (2008) Sensing meets mobile social networks: the design, implementation and evaluation of the CenceMe application. In: Proceedings of the 6th ACM conference on embedded network sensor systems (SenSys '08), pp 337–350, 2008
26. Beach A, Gartrell M, Akkala S, Elston J, Kelley J, Nishimoto K, Ray B, Razgulin S, Sundaresan K, Surendar B, Terada M, Han R (2008) WhozThat? evolving an ecosystem for context-aware mobile social networks. IEEE Netw 22(4):50–55
27. Dong Z-B, Song G-J, Xie K-Q, Wang J-Y (2009) An experimental study of large-scale mobile social network. In: Proceedings of the 18th international world wide web conference (WWW), 2009

28. Hui P, Chaintreau A, Scott J, Gass R, Crowcroft J, Diot C (2005) Pocket switched networks and the consequences of human mobility in conference environments. In: Proceedings of the 2005 ACM SIGCOMM workshop on delay-tolerant networking (WDTN ’05), 2005
29. Hui P, Crowcroft J (2007) How small labels create big improvements. In: Proceedings of international workshop on intermittently connected mobile ad hoc networks in conjunction with IEEE PerCom, 19–23 March 2007
30. Daly EM, Haahr M (2007) 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
31. Hui P, Crowcroft J, Yonek E (2008) 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), 2008
32. Gao W, Li Q, Zhao B, Cao G (2009) Multicasting in delay tolerant networks: a social network perspective networks. In: Proceedings of the 10th ACM international symposium on mobile ad hoc networking and computing (MobiHoc ’09), 2009
33. Zhang Y, Zhao J (2009) Social network analysis on data diffusion in delay tolerant networks. In: Proceedings of the 10th ACM international symposium on mobile ad hoc networking and computing (MobiHoc ’09), 2009
34. Bulut E, Szymanski BK (2010) Friendship based routing in delay tolerant mobile social networks. In: Proceedings of IEEE global telecommunications conference (GLOBECOM), 2010
35. Ying Z, Xu B, Shi X, Wang Y (2012) A survey of social-based routing in delay tolerant networks: positive and negative social effects. IEEE Commun Surv Tutor (to appear)
36. Mislove A, Marcon M, Gummadi KP, Druschel P, Bhattacharjee B (2007) Measurement and analysis of online social networks. In: Proceedings of the 7th ACM SIGCOMM conference on internet measurement (IMC ’07), 2007
37. Krebs VE (2002) Mapping networks of terrorist cells. Connections 24(3):43–52
38. Hossmann T, Legendre F, Spyropoulos T (2009) From contacts to graphs: pitfalls in using complex network analysis for DTN routing. In: Proceedings of the 28th IEEE international conference on computer communications (INFOCOM’09) workshops, pp 260–265, 2009
39. Hossmann T, Spyropoulos T, Legendre F (2010) 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
40. Hui P, Yoneki E, Chan SY, Crowcroft J (2007) Distributed community detection in delay tolerant networks. In: Proceedings of ACM SIGCOMM, workshop (MobiArch’07), 2007
41. Yoneki E, Hui P, Crowcroft J (2007) Visualizing community detection in opportunistic networks. In: Proceedings of ACM MobiCom workshop on challenged networks (CHANTS), 2007
42. Hui P, Yoneki E, Crowcroft J (2009) Identifying social communities in complex communications for network efficiency. In: Proceedings of the 1st international conference on complex sciences: theory and applications (Complex), 2009
43. Chan SY, Hui P, Xu K (2009) Community detection of time-varying mobile social networks. In: Proceedings of the 1st international conference on complex sciences: theory and applications (Complex), 2009
44. Hillery GA (1955) Definitions of community: areas of agreement. Rural Sociol 20(4):111
45. McMillan DW, Chavis DM (1986) Sense of community: a definition and theory. J Community Psychol 14(1):6–23
46. Perkins DD, Florin P, Rich RC, Wandersman A, Chavis DM (1990) Participation and the social and physical environment of residential blocks: crime and community context. Am J Community Psychol 18:83–115
47. Newman MEJ (2006) Modularity and community structure in networks. Proc Nat Acad Sci 103(23):8577
48. Okasha S (2005) Altruism, group selection and correlated interaction. Br J Philos Sci 56(4):730–725
49. Newman MEJ (2004) Detecting community structure in networks. Eur Phys J B 38(2):321–330

50. Girvan M, Newman MEJ (2002) Community structure in social and biological networks. Proc Natl Acad Sci 99(12):7821–7826
51. Newman MEJ (2004) Fast algorithm for detecting community structure in networks. Phys Rev E 69:066133
52. Danon L, Duch J, Arenas A, Daz-guilera A (2005) Comparing community structure identification. J Stat Mech Theory Exp 9008:09008
53. Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E (2008) Fast unfolding of community hierarchies in large networks. J Stat Mech 10:P10008
54. Everett MG, Borgatti SP (1998) Analyzing clique overlap connection. Connections 21(1): 49–61
55. Nguyen NP, Dinh TN, Tokala S, Thai MT (2011) Overlapping communities in dynamic networks: their detection and mobile applications. In: Proceedings of the 17th ACM annual international conference on mobile computing and networking (MobiCom '11), 2011
56. Marsden PV (2002) Egocentric and sociocentric measures of network centrality. Soc Netw 24(4):407–422
57. Freeman LC (1977) A set of measures of centrality based on betweenness. Sociometry 40(1):35–41
58. Freeman LC (1979) Centrality in social networks: conceptual clarification. Soc netw 1(3): 215–239
59. Newman MJ (2005) A measure of betweenness centrality based on random walks. Soc netw 27(1):39–54
60. Mei A, Morabito G, Santi P, Stefa J (2011) Social-aware stateless forwarding in pocket switched networks. In: Proceedings of the 30th IEEE conference on computer communications(INFOCOM) mini-conference, 2011
61. Leguay J, Friedman T, Conan V (2005) DTN routing in a mobility pattern space. In: Proceedings of the 2005 ACM SIGCOMM workshop on delay-tolerant networking (WDTN '05), 2005
62. McPherson M, Smith-Lovin L, Cook JM (2001) Birds of a feather: homophily in social networks. Annu Rev Sociol 27:415–444
63. Wu J, Wang Y (2012) Social feature-based multi-path routing in delay tolerant networks. In: Proceedings of the 31st IEEE international conference on computer communications (IEEE INFOCOM 2012)
64. Zhao L, Li F, Zhang C, Wang Y (2012) Routing with multi-level social groups in mobile opportunistic networks. In: Proceedings of IEEE global telecommunications conference (GlobeCom 2012), Anaheim, California
65. Gao W, Cao G (2011) User-centric data dissemination in disruption tolerant networks. In: Proceedings of the 30th IEEE conference on computer communications (INFOCOM), 2011
66. Fabbri F, Verdone R (2011) A sociability-based routing scheme for delay-tolerant networks. EURASIP J Wirel Commun Netw 2011:13

Chapter 13
Routing Protocols in Infrastructure-Less Opportunistic Networks

**Sanjay Kumar Dhurandher, Deepak Kumar Sharma,
Isaac Woungang and Shruti Bhati**

Abstract Opportunistic Networks (OppNets) remain one of the most researched extensions of Mobile Ad hoc Networks (MANETs) in the present times. Although characterized by sparse connectivity and indefinite paths, OppNets are highly scalable and flexible. They are quite different from the traditional MANETs as there is never a fixed path between the source and the destination, while in MANETs first a path is established between the sender and the receiver and then the message transfer takes place. They are considered as the subclass of Delay Tolerant Networks (DTN) as they support various features of DTNs. The delivery of message is reliant on hop-to-hop distribution through the network. A varying number of routing schemes have been proposed in the literature that cater to the efficient delivery of messages for various scenarios, movement models, and infrastructure. In this chapter, we present the recent related work focusing on the already existing routing protocols of infrastructure-less OppNets for various scenarios along with their relative advantages and disadvantages.

Keywords Opportunistic networks · Opportunistic routing · Infrastructure-less protocols · Delay tolerant networks · Direct transmission · First contact · Epidemic routing · Spray and wait · Spray and focus · Adaptive spray and wait ·

S. K. Dhurandher · S. Bhati
CAITFS, Division of Information Technology, Netaji Subhas Institute of Technology,
University of Delhi, New Delhi, India
e-mail: dhurandher@gmail.com

S. Bhati
e-mail: sb.shrutibhati@gmail.com

D. K. Sharma
Division of Computer Engineering, Netaji Subhas Institute of Technology, University of Delhi,
New Delhi, India
e-mail: dk.sharma1982@yahoo.com

I. Woungang (✉)
Department of Computer Science, Ryerson University, Toronto, Canada
e-mail: iwoungan@scs.ryerson.ca

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_13,

PRoPHET · PRoPHET+ · HiBOp · Content encounter probability-based message forwarding (CEMPF) · Robust proactive routing protocol · Repository-based forwarding protocol · MaxProp · Context-aware routing (CAR) · Meetings and visits (MV), Network coding

13.1 Introduction

The ubiquitous and ever-spreading nature of wireless and mobile technologies has made connectivity a daunting problem that requires an immediate solution. Sparse network density and highly mobile node behavior further complicates the matter. An opportunistic network [25] is a suitable solution for the aforementioned scenarios. It is one of the most interesting and recent evolutions of *Mobile Ad hoc Networks*. They utilize hop-by-hop distribution of messages to deliver it to the destination. The traditional routing algorithms used for MANETs and the Internet are not applicable here as they first establish a path between the source and the destination before the actual message transfer, which is not possible in case of OppNets [37, 7].

OppNets are enabled to deliver messages even when there is no connected path between the source and the destination. There is no previous knowledge of a suitable path or appropriate next node. The message is transferred from one node to another assuming that it will take the message near to the destination and eventually deliver it to the destination. As mentioned earlier, unlike in MANETs, high mobility and frequent disconnections are considered as norms. These characteristics are utilized for better dissemination of messages through the network. The protocols designed for such kind of networks utilize the opportune meetings between the nodes to deliver the messages to the destinations. Various parameters are used by various protocols to judge the suitability of a node to carry forward a particular message [8, 9, 23, 26, 30, 35, 47, 48, 50]. It utilizes whatever resources it has at its dispense to choose a node that can carry the message further closer to the destination [11, 16].

One key characteristic of opportunistic networks is that they are essentially delay tolerant in nature as they can handle large delays in message delivery from the source to the destination. OppNets share similar routing algorithms as used in *Delay Tolerant Networks* [13, 49]. They exhibit a store-carry and forward approach. If a suitable node is not found, the node simply stores the message and carries it through the network until a better node or destination is found [11]. Opportunistic network solves the problem of disrupted and sparse connectivity using the hop-by-hop approach.

The routing protocols in OppNets are divided into two categories, namely, the infrastructure-based and the infrastructure-less protocols [34].

(a) *Infrastructure-based protocols*: These protocols make use of some form of infrastructure to opportunistically forward messages. Base stations and access points are often involved in the routing and forwarding of the messages.
(b) *Infrastructure-less protocols*: These protocols are best suited for the flat ad hoc networks. They only make use of the mobility of the nodes and the contact

opportunity between them in order to route the messages. They make no previous assumptions about the network topology and all the nodes behave the same in the network and are given the same priority.

In this chapter, we will focus only on the infrastructure-less routing protocols for OppNets. The rest of the chapter is organized as follows. In Sect. 13.2, the related work in the area of routing protocols for infrastructure-less OppNets is presented with the help of examples. This is followed by a tabular comparison of the protocols based upon some common parameters. In Sect. 13.3, we enlist our conclusive remarks on the chapter.

13.2 Background and Related Work

In this section, we present a brief overview of the significant concepts on routing protocols for infrastructure-less Oppnets, namely, Direct Transmission [40], First Contact [18], Epidemic Routing [45], Spray and Wait [41], Spray and Focus [42], Adaptive Spray and Wait [29], PRoPHET [27], PRoPHET+ [17], HiBOp [2], Content Encounter probability-based message forwarding (CEMPF) [28], Robust Proactive Routing Protocol [21], Repository-Based Forwarding Protocol [15], MaxProp [3], Context-aware routing (CAR) [32], Meetings and Visits (MV) [4], Network Coding [46], along with their relative advantages and disadvantages.

13.2.1 Direct Transmission

In this protocol [40], the source node does not forward the message to the intermediate nodes, but stores it in the buffer until it comes into direct contact with the destination node. On encountering the destination node, the message is directly given to the destination. It is the simplest imaginable routing algorithm. Direct transmission is essentially a single-copy scheme.

Advantages

1. This protocol utilizes minimum network resources, and cannot suffer from the problems of network clogging.
2. The protocol is very simple and fairly easy to deploy.

Disadvantages

1. It is likely to suffer from heavy delays as source node may not encounter the destination node for longer periods of times.
2. If a node failure occurs, the message can be lost since there is only one copy available in the network.

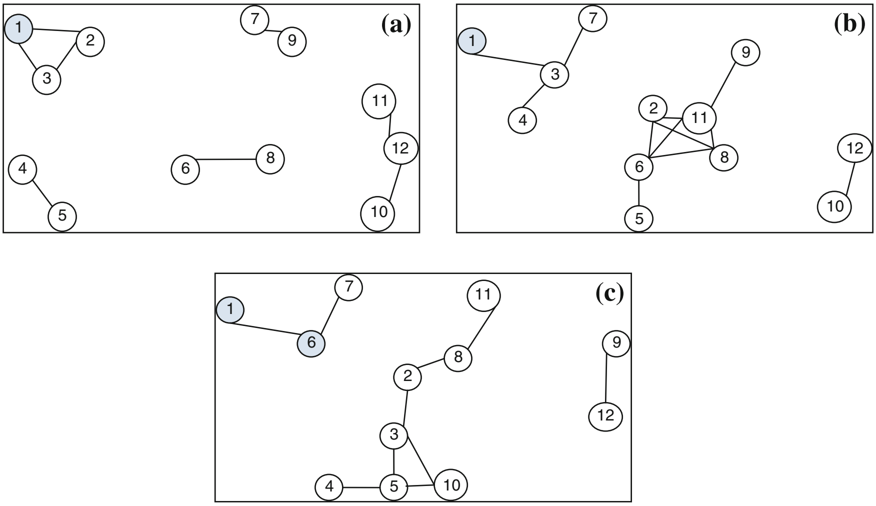


Fig. 13.1 Example simulation of direct transmission

Consider an example network scenario consisting of 12 nodes numbered from 1 to 12. Due to the mobility of nodes their position changes with time. Three different snapshots of the network with different configuration of nodes at three different times are shown in Fig. 13.1a–c respectively. The nodes which are in radio range of each other are shown connected by a line.

Figure 13.1 shows a simulation of a test run of direct transmission protocol. The message starts at node 1. Node 1 encounters several different nodes but delivers the message only to node 6 which is the destination as shown in Fig. 13.1c.

13.2.2 *First Contact*

First Contact [18] is a routing scheme that does not predict, utilize, or assume any properties of the network or the nodes. The carrier node forwards the message to any node that comes into contact with it. If no node is in its range, it stores the message and waits until a node becomes available. The protocol is rife with heavy delays and message dropping.
Consider the same example network scenario mentioned above:

Advantages

1. It does not make any assumptions about the network and thus can be implemented easily.
2. This scheme can be used for multicast messages.

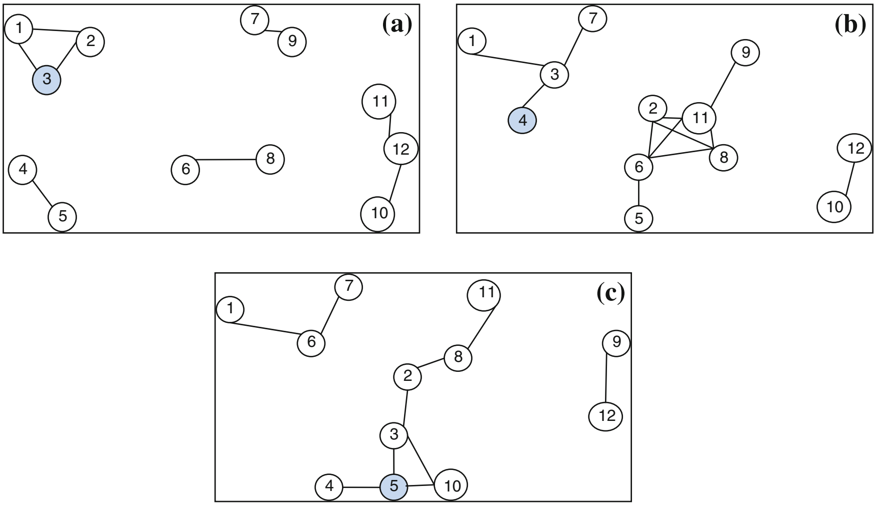


Fig. 13.2 Example simulation of first contact

Disadvantages

1. The scheme can lead to huge message delivery delays.
2. Indiscriminate spreading of messaging can lead to network clogging as well as packet dropping which is highly unacceptable.

Consider the same example network scenario mentioned above:

In Fig. 13.2, it can be observed that the message is passed on to the node that is encountered first by the message carrier. The message originates at node 1 (shown in Fig. 13.2a). Node 3 is the one that is encountered first; hence the message is transferred to it. A similar process is done in Fig. 13.2b, c.

13.2.3 *Epidemic Routing*

This protocol [45] makes minimal assumptions about the network and is guaranteed to deliver the message to the destination. The protocol aims to distribute messages to other nodes within connected portions of the network. These nodes then come into contact with another portion of the network and the message spreads like a disease (epidemic). Such a message transmission results in high probability of message delivery.

Multiple copies of the message are created in the network simultaneously. The only requirement for the protocol is a pair-wise connectivity for the eventual delivery of the message. Every message in the network has a unique message ID. Each node maintains a *Summary Vector (SV)* of the messages it produces and buffers on behalf of other nodes. An anti-entropy session is started by the initiating node (the one with the smaller identifier) which then shares its *SV* with other nodes. The *SV* of the first node is logically ANDed with negation of the *SV* of the second node. The difference in the message sets is found between the two nodes, and the nodes exchange those messages with each other which they do not have with them. The receiving node has the authority and the autonomy to decide whether it wishes to accept a message or not.

In classic epidemic routing, each message is associated with a unique message identifier, a hop count, and an optional Acknowledgment (ACK) request. The message identifier is a unique 32-bit number. This identifier is a concatenation of the host's ID and a locally generated message ID (16 bits each). The hop count in a message limits the maximum number of opportunistic exchanges that can take place for it. It is similar to the Time-to-live (TTL) field in the IP packets. Larger hop counts spread the message more in the network and smaller hop counts do the opposite. A hop count of one delivers the message directly to the destination only. Larger hop counts reduce the average delivery time, but also result in more resource consumption.

The protocol reduces the aggregate system resources consumed by limiting the number of hops of every message and the buffer space at every node in the network. Each node sets a maximum message buffer limit, thus limiting the number of messages that can be saved at any instance. Various policies can be implemented such as First-in-first-out (FIFO), priority ordering, etc., to implement the memory management.

Advantages

1. The protocol makes minimum assumptions about the network topology and it is fairly easy to deploy and understand.
2. Of all the opportunistic network protocols, epidemic has the least overhead in terms of calculations for determining the next hop.
3. The message delivery probability is very high in this protocol.

Disadvantages

1. The memory and resource consumption is very high in this protocol, as the message is passed to all the nodes indiscriminately.
2. A considerable amount of computation occurs at every node before exchanging the messages even in the case of nodes that might have the same messages. Hence, some amount of memory is wasted.
3. Even when a message is received at the destination, some nodes still continue passing on the messages, which wastes resources.

Consider the same example network scenario mentioned above:

Figure 13.3 shows a simulation for the epidemic routing. Node 1 which is the originator of the message delivers the message to every other node it encounters and finally reaches to node 10 which is the destination node as shown in Fig. 13.3c. It can be noted that the nodes already carrying the message, do not get a second copy of the message.

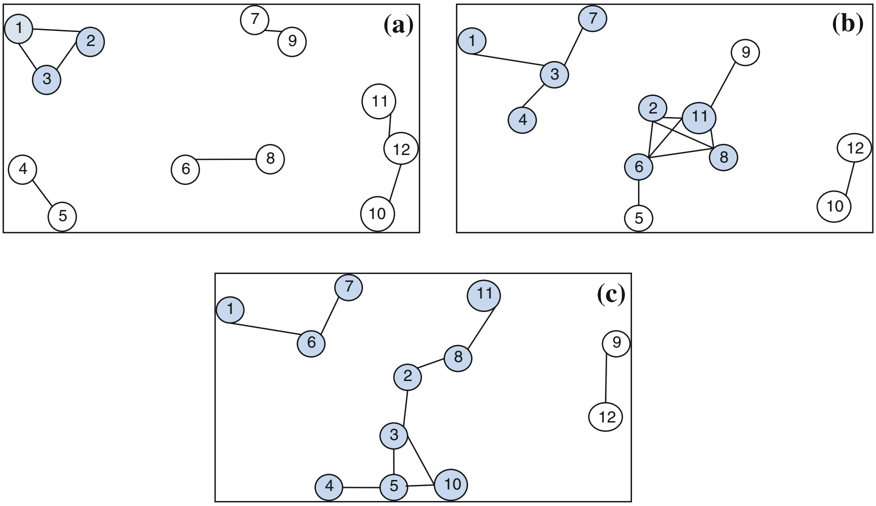


Fig. 13.3 Example Simulation of epidemic routing

13.2.4 Spray and Wait

This protocol [41] is an extension of the epidemic routing protocol. It aims to reduce the overhead of flooding which often causes network congestions and indiscriminate usage of the network resources. One way of doing so is to only forward a copy with some probability. The probability is the utility of every node based on the timer indicating the time elapsed since the two nodes last encountered the node maintaining the record. These are indirectly the relative node locations.

There are two phases in the forwarding process—*Spray phase* and *Wait phase* as mentioned in [41].

Spray phase—Every message generated is spread randomly to L relay nodes, i.e., L copies of message are created.

Wait phase—If the destination is not found in spray phase, the nodes then wait for direct transmission.

When enough copies have been spread to guarantee that the message will be delivered, direct transmission is then started. The protocol is a tradeoff between single and multi-copy message schemes [41]. The protocol combines the speed of epidemic routing with the simplicity of direct transmission. However, the definition above does not specify any value for L. Several heuristics may be applied to arrive at a suitable value for efficient delivery of the message. It is to be noted however that a smaller value of L makes the protocol similar to direct contact and a larger value of L makes it similar to the flooding. Hence an optimum value must be selected to minimize delay and resource consumption and maximize the probability of delivery.

Binary Spray and Wait—It is a variation of the spray and wait protocol. The source starts initially with L copies of the message. The source or any relay node that has

$n > 1$ message copies hands over $\lfloor n/2 \rfloor$ copies to any other node (without message copy) in the network and keeps the remaining copies $\lceil n/2 \rceil$ with itself. When only one copy is left, the relay node waits until it encounters the destination. Under the assumption that the node mobility is independent and identically distributed, this has been the most optimum method to route the messages.

Advantages

1. The protocol reduces the memory inefficiencies of epidemic routing by limiting the amount of flooding caused in the network.

Disadvantages

1. Although L is chosen to limit flooding, the protocol still suffers from delays and resource consumption issues.
2. The message delivery probability depends highly on the value of L chosen, which is assumed on the basis of the network parameters.

Consider the same example network scenario as mentioned above.

Figure 13.4 shows the simulation of binary *Spray and Wait* for $L = 5$. In Fig. 13.4a, node 1 (which is the sender node) passes on the message to only two nodes (node 2 and node 3). It is still left with two more copies which are distributed in Fig. 13.4b, c. Node 2 and node 3 each get only two copies of the message.

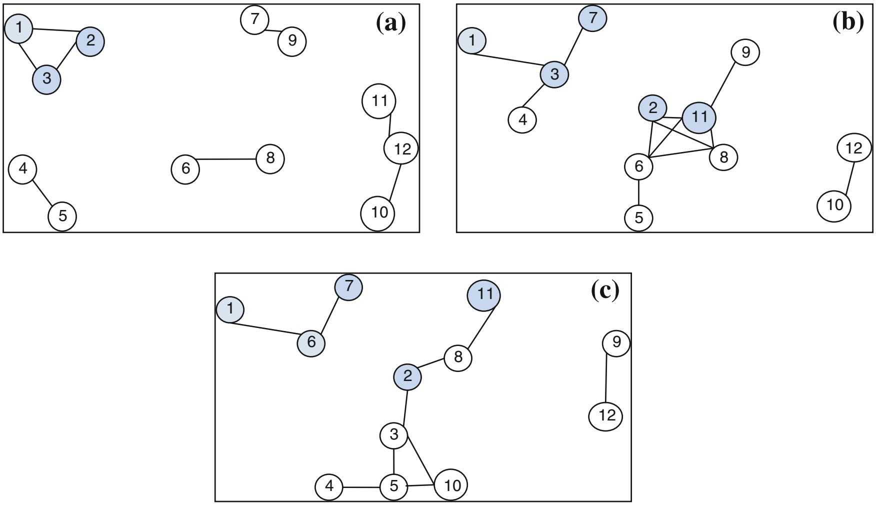


Fig. 13.4 Example simulation of binary spray and wait

13.2.5 *Spray and Focus*

This protocol [42] overcomes the shortcomings of simple spraying algorithms and is better than the flooding algorithms. Spraying schemes must have high mobility in order to perform well in any given network. In spray and wait, the message copies are spread quickly to the node's immediate neighborhood but very few of those nodes carrying a copy ever see the destination node. This problem is solved by using a better single-copy scheme. The scheme again has two phases:

Spray phase—When a new message is generated at the source node, it creates L forwarding tokens for this message. A node which has the forwarding tokens can forward the additional copy of the message to the other nodes. Each node maintains a summary vector of all the messages it carries. When two nodes meet, these vectors are exchanged. When a node meets another node with no message copy, the first node copies the message to the second node along with $n/2$ forwarding tokens. When a node has only one forwarding token, then it forwards the message according to the *Focus phase*.

Focus phase—Unlike spray and wait, where in the wait phase the forwarding was done using direct transmission, the forwarding here is done based on some criterion. These decisions are based on a set of timers that record the time since the two nodes last saw each other. Node i maintains a timer $T_i(j)$ for every other node j, which records time elapsed since the two nodes last saw each other. When a node encounters the other, their timers are set to zero and increased with every clock tick.

Here, [42] the authors consider $T_m(D)$ to be the expected time a node takes to move a distance D under a given mobility model. Then if node a encounters a node b at a distance D_{ab}, then according to [42]

$$\text{set } T_i(j) = T_j(i) = 0 \text{ and,} \tag{13.1}$$

$$\begin{aligned} &\text{If} \quad T_b(j) < T_a(j) - T_m(D_{ab}) \quad \text{then,} \\ &\text{set} \quad T_a(j) = T_b(j) + T_m(D_{ab}) \end{aligned} \tag{13.2}$$

These timers are used as utility functions and a node having a higher value of the utility function is handed over the message in the *Focus phase*.

Advantages

1. Owing to refined selection criterion, the protocol has higher message delivery rates.
2. Fewer copies are spread into the network as compared to the spray and wait protocol.

Disadvantages

1. The protocol suffers from a larger overhead of resource consumption.
2. Protocol suffers in case of sparse networks as the time taken by nodes to meet will be greater and there will be fewer opportunities to forward the data.

13.2.6 Adaptive Fuzzy Spray and Wait

This protocol [29] has been proposed as an improvement over the popular spray-based routing schemes. It smartly integrates the overheads and buffer management policies into an adaptive protocol that includes local *network parameters estimation*. The protocol has a clear distinction between forwarding and dissemination strategies. While forwarding attempts to deliver only one copy to the best node amongst the neighbors, dissemination attempts to deliver message by diffusing it through the entire network.

The routing scheme utilizes the various advantages of spraying approach like their utility over flat ad hoc networks, less overhead of complexity, and processing decisions. The algorithm can be summarized in the following few steps:

1. The node on encountering other nodes divides the values of L by 2 and updates it in the message before passing it on.
2. The node passes on all the copies to the nodes it encounters except the last copy which is passed on as direct transmission.
3. The messages in the buffer are sorted by a priority decided by a *Fuzzy decision making* function.
4. When the buffer is full, the messages are dropped according to the priority level, i.e., the oldest first.

The prioritization of the messages in the buffer is done to ensure that the messages that have been spread more in the network have less priority and to ensure that more messages get transmitted in the short contact by giving less priority to large messages. *Forward Transmission Count* and *Message Size* are two indicators that help in determining the prioritization quantity. The dropping policy used for the algorithm is random instead of drop least priority scheme so that it remains fair to all the messages.

Advantages

1. The scheme has better delivery performance than simple spray-based techniques.
2. Appropriate and fair buffer management schemes used in this protocol avoid the clogging of the network.

Disadvantages

1. Large messages might get delayed if there are a higher number of small messages.
2. If all the messages are of approximately the same size and a constant indicator of size is used, the size of message becomes irrelevant to priority.

13.2.7 PRoPHET

Probabilistic Routing Protocol using History of Encounters and Transitivity (*PRoPHET*) [27] assumes that instead of moving randomly the nodes in a network move in a predictable fashion wherein patterns are likely to repeat themselves. This information can be used to improve the routing performance. Each node before sending a message, calculates a probabilistic metric called *Delivery Predictability* for each known destination in the form of vectors. This *Delivery Predictability* is assumed to be $P(a, b) \in \{0, 1\}$, i.e., the probability of every node a to meet any other node b in the network. When two nodes meet, they exchange and update their *Delivery Predictability* vectors. Based on these vectors, messages are exchanged between the nodes.

The calculation of delivery predictabilities is done with the help of three equations as given in [27]:

$$P(a, b) = P(a, b)_{old} + (1 - P(a, b)_{old}) \times P_{init} \tag{13.3}$$

Equation 13.3 updates the *Delivery Predictability* whenever a node is encountered so that nodes that are encountered more often have a higher probability, where P_{init} is the predictability initialization constant.

If a pair of nodes does not encounter each other for a while the probability must be reduced and this is called aging. This reduction is done according to Eq. 13.4

$$P(a, b) = P(a, b)_{old} \times g^k \tag{13.4}$$

Here k is the number of time units that have elapsed since the last time the metric was aged and g is the aging constant that lies between 0 and 1.

The *Delivery Predictability* has a transitive property based on the observation that if node a frequently encounters node b and node b encounters node c, then node b is a good forwarder for node a's messages to node c. This transitivity is updated according to Eq. 13.5

$$P(a, c) = P(a, c)_{old} + (1 - P(a, c)_{old}) \times P(a, b) \times P(b, c) \times V, \tag{13.5}$$

where V is the scaling constant that decides the impact of transitivity on *Delivery Predictability*.

When two nodes meet, a message is transferred to the other node if the *Delivery Predictability* of the destination of the message is higher at the other node, otherwise it is kept with the current node.

Advantages

1. Simulation results of this protocol show that it has less message exchanges, less communication overhead, less delay, and higher delivery success rate as compared to the epidemic routing.
2. The protocol is highly suited to human mobility scenarios.

Disadvantages

1. Resource consumption occurs in terms of calculations occurring at each node.
2. Memory is needed to store the probability tables generated by the protocol.

3. Packets may be dropped consistently when forwarded to a few concentrated nodes due to FIFO queueing nature of PRoPHET.

13.2.8 PRoPHET + (An Adaptive PRoPHET-Based Routing Protocol)

The following protocol [17] is an extension of the PRoPHET protocol that uses only delivery predictabilities of the nodes to decide the next best carrier for the message. This scheme uses several other parameters such as *Buffer*, *Power*, *Bandwidth*, *Location*, and *Popularity* into consideration to reduce the packet loss and transmission delay while choosing the best next hop among various nodes.

The nodes are assumed to participate in routing without any malicious behavior. The different parameters are assigned certain weight and these values are then normalized to bring final value between 0 and 1. A brief discussion of the various parameters given in [17] is as follows:

1. *Buffer parameter*: All nodes define a threshold B_{thres} value which is computed as $B_{total} - B_{self}$. Here B_{total} is the total buffer size of the node and B_{self} is the amount of storage necessary for self-generated data. The sender node asks for B_{thres} to the neighboring node and then calculates the V_B which is the value of buffer parameter, by dividing the remaining buffer size minus the packet size of sender node over B_{thres}.
2. *Power parameter*: Data loss due to node failure or power failure at node is reduced using this parameter. To calculate V_P, i.e., the power parameter's value for rechargeable nodes, the following formula defined in [17] is used:

$$(\mathrm{P}remain - \mathrm{P}receive - \mathrm{P}send - \mathrm{P}threshold)/\mathrm{P}total \tag{13.6}$$

   where $Premain$ is the remaining power, $Preceive$ is the power required to receive a packet, $Psend$ is the power required to send a packet, and $Ptotal$ is the total power of the device.
3. *Bandwidth parameter*: In order to reduce the chances of unsuccessful packet delivery due to short contact periods between nodes, a faster bandwidth is required. This is of utmost need in faster mobility networks. In order to determine the bandwidth parameter's value of a node (V_A), the sender node sends data to its neighbors at maximum transmission rate. The neighbors then use their own maximum transmission rate divided by sender's maximum transmission rate to calculate V_A.
4. *Popularity parameter*: This parameter is used in order to eliminate those nodes that usually carry a significant portion of the total data over the network. If such nodes fail, that data may be lost forever. Hence, such nodes must be avoided. The popularity parameter value (V_O) is defined as N_t/M_t, where N_t is the number of transmissions the node has performed in a certain amount of time and M_t is the

maximum number of transmissions the node can perform in that same amount of time.

To calculate the *deliverability value* (V_D) of any node, a weight for each parameter value is assigned. The V_D value is calculated by the equation [17]:

$$\mathrm{V_D = W_B(V_B) + W_P(V_P) + W_A(V_A) + W_O(V_O) + W_R(V_R)} \tag{13.7}$$

where V_R represents the predictability value from *PRoPHET*, and W_B, W_P, W_A, W_O, and W_R are weights for buffer, power, bandwidth, popularity, and predictability value respectively.

Advantages

1. Packet dropping and loss are significantly reduced as compared to *PRoPHET* by monitoring the power and popularity of the node.
2. The protocol takes into account several parameters other than delivery predictability which strengthens the best node probability.

Disadvantages

1. Too many calculations need to be done at each node.
2. Uncooperative nodes can cause a problem for the sender.

13.2.9 History-Based Routing Protocol (HiBOp)

The History-Based Routing Protocol for Opportunistic Networks (HiBOp) [2] aims at effectively utilizing the context information of the node in order to decrease the overhead of flooding. The *Current Context (CC)* of a user is a snapshot of the local environment of the user (e.g., name, residential address, working address phone, date of birth, etc.). It is stored in the form of *Identity Tables (ITs)*.The user has complete autonomy on the information shared in the *ITs*. Nodes exchange ITs during neighbor discovery phase and build a *CC* for themselves using their neighbors *ITs*. New nodes exchange complete *ITs*, whereas stable nodes that have been around for a few *'Signaling Interval'* (constant time period) exchange only their *Node Identity (NIDs)*, which is a hash of the *ITs*.

Every node also stores a *History table* that stores values from the *ITs* seen by the node in the past. Every value has a *Continuity Probability* (P_c), *Heterogeneity (H) and Redundancy (R)* counters associated with them. To update the *History table* an intermediate *Repository table* is created which appends data to it at every *Flushing Interval*. Therefore, the *Continuity Counter* stores how many times that attribute has been seen in the *Current Context* during a *Flushing Interval*. *Heterogeneity* counter indicates how many different *ITs* carry the same attribute and the *Redundancy* counter indicates the number of times the attribute occurs in the same *IT*.

The forwarding process in HiBOp is made of three phases [2]:

1. *Emission Phase:* HiBOp injects the message into the network through flooding to an appropriate number of nodes by creating message replicas for reliability. The number of neighbors (K) to which the message is forwarded by the sender is calculated using a formula that takes into account the probability of delivering the message to the destination. The value of K should be minimized in order to minimize the *joint loss probability* below a certain threshold.
2. *Forwarding Phase*: It uses the node's mobility and contacts to take the message closer to the destination. It uses two quantities to determine the next best node which is described below:
   a. The forwarding of message to a certain node during its journey in the network is determined by the match between the sender information and the context information of the nodes. Matches are assigned certain weights representing the precision with which that attribute (belonging to a class) can identify the destination. The weights assigned to those classes are determined by using a function [2]:
   $$\mathrm{W_{i+1}} = \mathrm{W_i} + \mathrm{r_i} \times B \tag{13.8}$$
   where B is weight increase parameter and r_i is the maximum redundancy at the ith class.
   b. Delivery predictability of a node is also taken into account before passing on the message. At each node the delivery probability is calculated using node *IT*, its *CC* and its *History table.*
3. *Delivery Phase*: When an intermediate node finds the destination, the message is delivered to it and the process stops.

Advantages

1. This protocol stores the largest amount of context information among all context based protocols, so it can fully exploit the advantages of context information.
2. The protocol is very suitable for human mobility models [6, 14, 19, 24, 39] that generally follow a particular pattern.
3. The protocol reduces network clogging by drastically limiting the number of copies spread in the network.

Disadvantages

1. The *IT*, *CC*, *History table*, and *Repository table* require a large amount of memory space on every node.
2. The calculations done every time at every node can significantly reduce the amount of time left for message exchange.

13.2.10 Content Encounter Probability-Based Message Forwarding in Opportunistic Networks (CEPMF)

This protocol [28] uses the content of the messages to relay and deliver them to their destination. The node that generates the messages is called the *publisher* and the node that wants a message similar in content to the message being sent is called *subscriber*. This solves the problem where the address of the recipients is not known. If subscribers demand a message of a particular type of content, they spray the network with their interests all over the network in the form of *predicates* (i.e. a description of the content the subscribers wants). Every node in the network maintains predicate-utility tables for predicates they might have received during their journey. The utilities indicate the priority of the message type to be relayed first.

The entire protocol can be divided into two steps:

1. *Predicate propagation:* The *subscriber* spreads its interests into the network by spraying *predicates*. These are essentially "logical disjunction of conjunctions of elementary constraints over the values of individual attributes" [28]. The *Spray* and *Wait* scheme is used for sending the *predicates*. *Subscriber* generates L copies and hands over $L/2$ copies to first node it encounters. If only one copy is left, it starts the *Wait* step. In order to simulate a predictable movement and spraying each predicate, the *encounter probability ep* $\in (0, 1)$ is used. It calculates the *ep* between predicate relays and predicate senders [28].

Spray Phase: Each predicate is given an initial *ep* .This value is stored in the *ep*-table.

Wait Phase: If node carrying predicate with value *ep* does not meet the node that has the same predicate, then the value of *ep* is decreased exponentially using the formula [28].

$$ep_{new} = ep_{old} \times e^{-(t-t0)} \tag{13.9}$$

where ep_{new} and ep_{old} are new and old values of *ep*, t_0 is the last moment that predicate encounters with a same content request, and t is the current time.

If a node encounters another node that has content requirements as the predicate it carries, the value of *ep* is increased using the formula [28].

$$ep_{new} = ep_{old}(1 - ep_{old}) \times ep_{init} \tag{13.10}$$

where ep_{init} is the initial value of *ep*.

There might be cases where contents carried by different nodes may have some common *predicates*. In such a case the *predicates* are merged to increase buffer management.

2. *Message Forwarding*: Each node has a table carrying the message and is tagged to its *ep* value. At source node the value is zero. When a node transfers message to a node carrying the message's *ep* value in its table, the acknowledgment tells the source to stop sending until it is out of its range. In case of relay nodes, the message is transferred if a node has *predicate* value greater than the value in its own table. In this case an acknowledgment is sent and the time-to-live is reduced to half. In this

scheme, when the *subscriber* receives the message the relay nodes do not delete it because it is believed that there may be other nodes in need of it. Thus, it can be concluded that there is no single destination.

Advantages

1. The protocol can be safely used for multicast messaging situations.
2. It does not rely on the geographic location of the nodes and thus can be used in cases where GPS is not enabled.

Disadvantages

1. A node may still receive a particular message which it does not want, if its demands are partially similar to the predicates of the message.
2. There might be high privacy and risk issues associated with the content transparency.

Consider the same example network scenario as mentioned above. Here, predicate D is the predictability of the node for destination node (in this case node 10). A message that matches predicate D is relayed. The source has a value of zero. In Fig. 13.5a since node 3 has a value of 0.93, it is exchanged. In Fig. 13.5b node 4 has a value of 0.95, so it is again exchanged. However, in the last Fig. 13.5c node 5 has a value of 0.85 which is less than 0.95 and hence is not exchanged.

13.2.11 Robust Proactive Routing Protocol

This protocol [21] has been proposed as a proactive scheme to deliver messages in the highly disconnected scenarios present in OppNets. It is an adaptive protocol

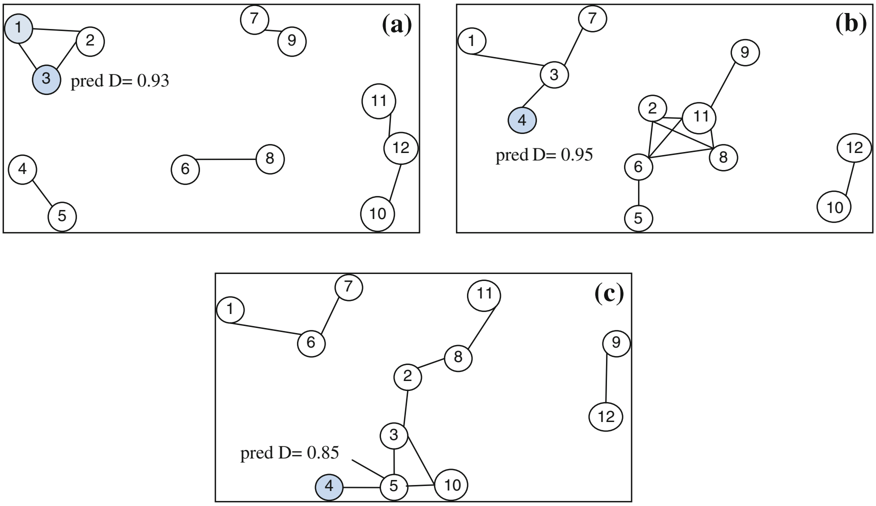


Fig. 13.5 Example simulation of CEMPF

that uses the contact opportunity between the nodes to determine the neighborhood information for each node and use that to deliver the messages. It is assumed that the nodes move in a predicable fashion and have the capability to recognize their surroundings. The basic principle is that a node can determine the predictability information and connectedness information using past history. The nodes do not forward data randomly, instead they have a well-studied and selected next hop node.

The nodes maintain information about future and past contacts in a connectivity table. This table is exchanged with every node in a connectivity summary vector format. When a node receives this vector from its neighbor, it compares it with its own summary vector and updates accordingly. This ensures that the table is completely up-to-date about the best forwarders for a packet. Nodes often change the structure of the table to include information for more than two nodes (relay nodes) for a single destination. Thus, a node has a variable choice when selecting a node for forwarding the message and can choose best node/nodes suited to the situation (Table 13.1).

Advantages

1. The following protocol is suited to Human mobility models since it can show better results for predictable movement.
2. It reduces the overhead of flooding by limiting the next hop for node and only selects the best candidate.

Disadvantages

1. There is considerable overhead in exchanging and storing tables at each node.

The example structure of the connectivity table used in this protocol is given below:

13.2.12 Repository-Based Forwarding Protocol (RFP)

This protocol [15] makes distinction between the types of nodes in the network and uses them to deliver the message to its destination. It argues that assuming all nodes to be *Mobile Nodes* is not correct in a model where nodes move in a predictable fashion (Human mobility model). Some nodes are likely to be in a place more often than others. Hence, commonly visited and shared locations are assigned to be *Fixed Nodes*. These are installed in the most frequently visited places and are different from the access points [15]. An acknowledgment mechanism is used so that the buffer space can be conserved in the *Fixed Nodes*.

The *Mobile Nodes* are characterized by the mobility pattern which in turn has a strong impact on their performance. Each node has a *character table (CT)* that defines the node's home location, communication range, ID, and type. Every node maintains a buffer of its messages. The message packet is made of two parts:

1. The payload containing the content of the message.
2. The header which is made of five fields, namely *Source ID*, *Target ID*, *Message ID*, *Current node ID*, and *Status flag* indicating whether the message can be *tsend*,

Table 13.1 Example connectivity table as shown in [21]

Destination node no.	First choice (node no's)	Connection establishment time	Connection end time	Predictability value (%)
1	4	T1s	T1e	70
2	3	T2s	T2e	50
4	4	T3s	T3e	90
9	9	T4s	T4e	50

waiting, *delivered*, or received. The *tsend* status indicates a new message waiting to be sent, *waiting* indicates a new message waiting to be collected, and the *delivered* status indicates that the message has been delivered to the destination.

When a node meets another node, they exchange their *CT* tables that describe their characteristics. Depending on the type, one of the nodes initiates the forwarding process. The forwarding process is further divided into two parts:

(1) *Message Dispatch*: Node initiating communication sends all messages as type *tsend* either directly (to the target node itself) or indirectly (i.e. through a *Fixed Node* which eventually forwards it).
(2) *Message Collect*: Node collects all the messages that have the target_ID set as the node itself. The status flag is checked to prevent from the same message being collected again.

The detailed algorithm used in both the phases can be referred from [15].

Advantages

1. Fixed nodes can decrease the delivery delays considerably.

Disadvantages

1. If fixed nodes fail, the entire network will be brought down.

13.2.13 MaxProp

This protocol [3] assumes that it has no prior knowledge about the network connectivity and uses the local information and opportune movement to select the next best hop for the message delivery. It uses several other mechanisms in order to increase the delivery rate. The protocol can be divided into three main components that are listed below:
1. *Estimating Delivery Likelihood*: The protocol aims to find the optimal delivery paths by constructing a directed graph of nodes which are connected by edges [3]. A variation of Dijkstra's algorithm is used to determine the shortest path out of available paths at any given point of time. Every node i keeps the probability of meeting with its peer j denoted by f_j^i. Initially, this value is $1/(|\mathrm{s}| - 1)$ where s

Table 13.2 Comparison of various routing protocols

Protocol	Algorithm					Assumptions		Simulation model	
	Number of message copies	Next hop selection method	Drawbacks	Delivery delay	Delivery ack	Buffer size available	Bandwidth capacity	Simulator used	Mobility model used
Direct transmission [40]	Single	The destination of the message itself	High delays and low delivery latency	High	None	Limited	Not mentioned	Custom discrete event driven	Random waypoint (RWP) [5]
First contact [18]	Single	The next node encountered	High amount of clogging in the network	High due to path loops	None	Not mentioned	Not mentioned	Own for DTN environment [10]	Own remote village city bus N/W
Epidemic routing [45]	Unlimited	Flooding	High resource (bandwidth, buffer) usage	Low	Not mentioned	Limited	Not mentioned	Monarch [43]	RWP
Spray and wait [41]	Limited (L)	Randomness	Random decision making	Medium	Not mentioned	Sufficient	Sufficient	Own	RWP, RD,RW
Spray and focus [42]	Limited (L)	Timer based probabilities calculation	High resource consumption	Medium	Not mentioned	Sufficient	Sufficient	Own	RWP, random Walk(RW)
Adaptive spray and wait [29]	Limited	Flooding	Distinction made in size of messages	Medium	Not mentioned	Limited. Buffer management	Sufficient	ONE [22]	RWP

(continued)

Table 13.2 (continued)

Protocol	Algorithm					Assumptions		Simulation model	
	Number of message copies	Next hop selection method	Drawbacks	Delivery delay	Delivery ack	Buffer size available	Bandwidth capacity	Simulator used	Mobility model used
PRoPHET [27]	Single	Probability obtained from previous meetings	Too much calculation overhead at each node	Medium	Not mentioned	Limited	Not mentioned	Own	Own
PRoPHET+ [17]	Single	Probability of previous meetings, buffer, power, bandwidth and popularity parameters	Uncooperative nodes can cause security & forwarding problems	Medium	Not mentioned	Limited. Managed extensively	Sufficient and evaluated before transferring message	DTNSIM [12]	iMote trace [38]
HiBOp [2]	Single	Identity tables and history table used to find context of nodes	High overhead of storing tables	Medium	Not mentioned	Limited and managed	Assumed infinite	Own	Community based (CB)

(continued)

Table 13.2 (continued)

Protocol	Algorithm					Assumptions		Simulation model	
	Number of message copies	Next hop selection method	Drawbacks	Delivery delay	Delivery ack	Buffer size available	Bandwidth capacity	Simulator used	Mobility model used
CEMPF [28]	Limited Flooding	Content of the message and predicates	Privacy and risk issues	High as content of messages are matched	Relay to relay as well as source to destination ack used	Limited	Sufficient	OMNET++ [33]	A hybrid of RWP and CB
Robust proactive routing protocol [21]	Single	Predictability and connectedness information	Overhead of exchanging, storing and updating tables	Medium	Not mentioned	Limited	Sufficient	Jist/SWANS [1]	Not mentioned
RFP [15]	Single	Type of node	Failure of fixed nodes	Medium	Used	Limited	Sufficient	Own	Human mobility model
MaxProp [3]	Single	Previous node meetings and finding best path using likelihood of meetings	In case of sparse network the no. of available paths might decrease drastically	Medium	Used	Unlimited (own) Limited (others). Buffer management Used	Sufficient	Own	Synthetic models of real data

(continued)

Table 13.2 (continued)

Protocol	Algorithm					Assumptions		Simulation model	
	Number of message copies	Next hop selection method	Drawbacks	Delivery delay	Delivery ack	Buffer size available	Bandwidth capacity	Simulator used	Mobility model used
CAR [32]	Single	DSDV and Delivery probability using context information	Overhead of table management and exchange	Medium	Not mentioned	Sufficient	Sufficient	OMNET++ [33]	Own Group-based mobility model [31]
MV Routing [4]	Single	Probability of visiting the region of the destination	Overhead in storing the mobility pattern of a node at regular intervals	Medium more than Epidemic	Not mentioned	Unlimited (own) Limited (others). managed by FIFO.	Sufficient	NS-2 [44]	Synthetic traces of node movement in geographic area
Network coding [46]	Limited	Flooding to the neighbors	Overhead in encoding and decoding of message	Not Mentioned	Not mentioned	Not mentioned	Not mentioned	Own (custom time-based simulator)	RWP

is the total number of nodes. When i encounters j again, this value is incremented by 1 and all the values are then re-normalized. These values are exchanged between two nodes at every meeting. A local node cost is calculated for each path to the destination using n nodes in between with the formula given in [3].

$$\mathrm{c}(\mathrm{i}, \mathrm{i}+1, \ldots, \mathrm{d}) = \sum\nolimits_{x=i}^{d-1} [1 - (\mathrm{f}_{x+1}^{x})] \tag{13.11}$$

The cost for a destination is the path's cost which is lowest amongst all. The packets are ranked with some priority among them. The highest priority packets are transmitted first, and the packets with lowest priority are deleted subsequently. If two packets have the same priority, then the packet with fewer hops is transferred first.
2. *Complementary Mechanisms*: This step describes the priority order in which messages are exchanged when two nodes discover each other. These orders are listed below:

(a) First, all messages concerning the neighbor node are transferred.
(b) Second, routing information is exchanged.
(c) Third, acknowledgments are transferred.
(d) Fourth, packets that are new or have not propagated enough in the network are given priority and passed on next. For this purpose, the buffer is split into two parts with packets having hop count below t hops on the one side and with more on the other side.
(e) Fifth, the remaining untransmitted packets are exchanged.

3. *Managing Buffer*: The protocol states that there is a difference between managing limited storage and limited transmission in that the packets sent once can be sent again [3]. The three reasons for which the packets might be dropped by a node from its buffer are that it might have been delivered, absence of sufficient bandwidth, or no copy has been delivered.

Advantages

1. MaxProp uses Dijkstra's algorithm to ensure that the lowest cost path is chosen so as to decreases the delivery latency.
2. Proper buffer management schemes lead to a lowered rate of packet dropping.

Disadvantages

1. The overhead of table exchange can decrease the effective time for message exchange.
2. The protocol is not suited for sparse networks, as it will not give a proper connected graph and thus will not satisfy the protocol criteria.

13.2.14 Context-Aware Routing (CAR)

In this protocol [32], the nodes are assumed to rely on their 'logical connectivity information' with other nodes. They are not aware about the location of the nodes, which are the recipients of the messages they are carrying with them. Proactive protocol, such as DSDV [36], is used to deliver the messages if the recipient node belongs in the same cloud. Otherwise, the relay nodes are chosen in such a manner that they present the highest delivery probabilities. These probabilities are calculated locally at every node from the context information. The *Context* is defined as attributes of a node that can help to make the delivery mechanism more efficient. When two hosts meet they not only exchange these probabilities but also the DSDV information.

One key feature of this protocol is that it does not use the available *Context* of a node as it is, instead it predicts the future values so as to make the information more 'realistic' [32].The process of prediction and evaluation of context can be summarized in a few steps.

1. *Calculation of self-delivery probabilities*: This essentially consists of predicting the future values of attributes that make up a node's context. Kalman filters [20] -based estimation is used for this process. This helps in reducing the overhead of updating tables at every meeting and the values can be predicted now. From the history of values a prediction model is derived which is represented by a set of vectors.
2. *Table*: Every node stores a table of the best next hop and the delivery probability for all the known destinations. It is important to note that only one row corresponds to every destination.
3. *Local prediction of delivery probabilities at intervals*: Prediction process starts during temporary disconnections and is carried out until there is a guarantee for certain accuracy. Furthermore, updating other nodes is only done if the node evaluates and realizes that the sampled context information values are accurate.

The context information is represented as a set of attributes $(X_1, X_2, \ldots X_n)$. The idea is to maximize every attribute and figure out a *Utility function* represented by U$(x_1, x_2, \ldots x_n)$. A weight w is assigned with every attribute and the resulting function arrived is [32]:

$$maximize\{f(U(x_i)) = \sum_{i=1}^{n} w_i U_i(x_i)\} \tag{13.12}$$

The weights are themselves made adjustable and dynamic using adaptive weights a_i. Thus the function becomes [32]:

$$maximize\{f(U(x_i)) = \sum_{i=1}^{n} a_i(x_i) w_i U_i(x_i)\} \tag{13.13}$$

The $ai(xi)$ is a composite value. It is made of three essential parameters, namely [32]:

a. The range of parameters.
b. The predictability of the context information.
c. The availability of context information.

$$a_i(x_i) = a_{range}(x_i) \times a_{predictability}(x_i) \times a_{availability}(x_i) \tag{13.14}$$

4. *If new message recipient*: If a node encounters a message whose recipient does not feature in its table, the message is passed on to the most mobile node in the cloud so as to ensure that it reaches its destination.
5. *Message exchange:* The message is passed on to a node on meeting, which has higher probability of delivering the message.

Advantages

1. The protocol is uninfluenced by the unavailability of GPS.
2. The proactive approach highly reduces the overhead of prediction calculation and can deliver message faster if dense networks are present.

Disadvantages

1. The overhead of table exchange, updating, and maintenance can severely reduce the performance.
2. In the absence of a proper buffer management scheme, the messages may be lost.

13.2.15 Meetings and Visits (MV)

This routing protocol [4] uses the same pair-wise message exchange principle as *Epidemic Routing*, but improves on the method used to determine which messages to transmit. Instead of flooding its neighbors, each node uses observation data on the *meetings* between nodes and *visits* to locations (hence the name *MV*) to compute a delivery probability for every other node in the network. The past frequencies are used to rank each message in a peer's buffer according to the likelihood of delivering a message through a path of meetings and locations. It uses the node's motion patterns to increase the efficiency of message delivery. When two nodes meet, the summary vectors contain not only the message identifiers but also the computed delivery probability. Nodes compare their own and their pair's values, and only request messages for which their probability is higher. These messages are then erased from the source node, preventing message duplication.

Advantages

1. The protocol is highly suited to human mobility scenarios as mobility patterns of nodes are stored for routing of the messages.
2. It uses the techniques from robotic control to obtain high-quality approximations for the optimal solution.

3. It also limits the number of hops that are required, by calculating an estimation of delivery likelihood assuming an infinite buffer at each peer.

Disadvantages

1. As it uses First-in-first-out (FIFO) for buffer management at nodes, packets may get dropped consistently when forwarded to a concentrated node having a limited amount of buffer space.
2. Considerable overhead occurs in storing the mobility pattern of nodes at regular time intervals.

13.2.16 Network Coding

This protocol presents [46] an approach in which a message is encoded into another format before transmission. Additional information is embedded within the coded blocks such that the original message can be successfully reconstructed with only a certain number of the coded blocks. The intermediate nodes not only forward but can also combine packets using a given invertible function before forwarding to limit the message flooding. At the receiver side, as compared to the replication-based schemes which rely on successful delivery of each individual data block, this scheme considers the successful delivery of a block only when the necessary number of blocks is received to reconstruct the original data. These blocks can be just a small portion of the total number of blocks transmitted. As an example, in a communication involving two nodes (A and B) and an intermediate node C, in which node A sends a message x and node B sends a message y, node C can combine them into a single message $w = x \oplus y$, which it then broadcasts to both nodes. Thus node C, rather than sending two different packets for "x" and "y", respectively, broadcasts a single packet containing "x" or "y". Each node, when receiving message w can decode it using its own sent message to recover the other, halving node C's transmission needs. Network coding-based routing outperforms flooding, as it is able to deliver the same information with a fewer number of messages injected into the network.

Advantages

1. This scheme is more robust against packet losses than replication-based schemes when network connectivity is extremely poor, as it considers the successful delivery of a block only when the necessary number of blocks is received to reconstruct the original data.
2. The number of transmissions is reduced in this approach, and consequently the packet delivery ratio is much higher than the probabilistic forwarding both in dense mobile networks and sparse networks.

Disadvantages

1. When the network is well connected, this scheme is less efficient due to additional information embedded in the code blocks.

2. This method leads to additional processing power and memory requirements due to the encoding and decoding process.

Table 13.2 presents a comparison of the various routing protocols with respect to certain parameters.

13.3 Conclusion

In this chapter, we have given a brief outline of a number of routing protocols for infrastructure-less OppNets. These protocols have been analyzed and compared on the basis of their advantages and disadvantages with respect to a variety of parameters. While this chapter has not covered all the routing protocols, it attempts to provide an overview of the research techniques in this area by investigating a few key protocols. This study leads to the identification of some critical and explicit characteristics of each protocol along with their areas of application. A tabular comparison has also been provided based upon certain parameters related to all the protocols. A few main concerns that are reflected in almost all protocols are delivery latency, packet dropping and packet loss, memory management, computation, and storage overhead. The discussed protocols in this chapter implement different techniques to abate the inefficiencies caused due to these factors.

References

1. Barr R, Haas ZJ (2005) Jist: An efficient approach to simulation using virtual machines. Software practice & experience. Wiley, New York, pp 539–576
2. Boldrini C, Conti M, Iacopini I, Passarella A (2007) HiBOp: a history based routing protocol for opportunistic networks. In: Proceedings of IEEE international symposium on world of wireless, mobile and multimedia networks, 2007 (WoWMoM 2007), Espoo, Finland, 18–21 June 2007, pp 1–12
3. Burgess J, Gallagher B, Jensen D, Levine BN (2006) Maxprop: routing for vehicle-based disruption-tolerant networks. In: Proceedings of 25th IEEE international conference on computer communications (INFOCOM 2006), Barcelona, Spain, 23–29 April 2006, pp 1–11
4. Burns B, Brock O, Levine BN (2005) MV routing and capacity building in disruption tolerant networks. In: Proceedings of IEEE 24th annual joint conference of the IEEE computer and communications societies (INFOCOM 2005), vol 1, Miami, FL, USA, 13–17 March 2005, pp 398–408
5. Camp T, Boleng J, Davies V (2002) A survey of mobility models for ad hoc network research. Wireless Commun Mobile Comput (WCMC): Spec Issue Mobile Ad Hoc Netw Res Trends Appl 2(5):483–502
6. Chaintreau A, Hui P, Crowcroft J, Diot C, Gass R, Scott J (2006) Impact of human mobility on the design of opportunistic forwarding algorithms. In: Proceedings of 25th IEEE international conference on computer communications (INFOCOM 2006), Barcelona, Spain, 23–29 April 2006, pp 1–13
7. Chen L-J, Hung Yu C, Tseng C, Chu H, Chou C (2008) A content-centric framework for effective data dissemination in opportunistic networks. IEEE J Sel Areas Commun 26(5):761–772

8. Conti Marco, Kumar Mohan (2010) Opportunities in opportunistic computing. IEEE Comput J Mag 43(1):42–50
9. Conti M, Crowcroft J, Giordano S, Hui P, Nguyen HA, Passarella A (2009) Routing issues in opportunistic networks. In: Miranda H, Rodrigues L, Garbinato B (eds) MiNEMA state-of-the-art book. Springer, Berlin
10. Delay Tolerant Network Research Group (DTNRG), http://www.dtnrg.org/
11. Dhurandher SK, Sharma DK, Woungang I, Chao HC (2011) Performance evaluation of various routing Protocols in opportunistic networks. In: Proceedings of IEEE GLOBECOM workshop 2011, Houston, Texas, USA, 5–9 Dec 2011, pp 1067–1071
12. Ekman F, Keränen A, Karvo J, Ott J (2008) Working day movement model. In: Proceeding of the 1st ACM SIGMOBILE workshop on mobility models (MobiHoc 2008), Hong Kong SAR, China, 27–30 May 2008, pp 33–40
13. Fall K (2003) A delay-tolerant network architecture for challenged internets. In: proceedings of ACM SIGCOMM 2003, Karlsruhe, Germany, 25–29 Aug 2003, pp 27–34
14. Goodman DJ, Borras J, Mandayam NB, Yates RD (1997) INFOSTATIONS: a new system model for data and messaging services. In: IEEE vehicular technology conference 1997(VTC'97), vol 2, May 1997, pp 969–973
15. Greede A, Allen SM, Whitaker RM (2009) RFP: repository based forwarding protocol for opportunistic networks. In: Proceedings of third international conference on next generation mobile applications, services and technologies (NGMAST '09), Cardiff, Wales, 15–18 Sept 2009, pp 329–334
16. Huang C-M, Lan K-C, Tsai C-Z (2008) A survey of opportunistic networks. In: Proceedings of the 22 nd international conference on advanced information networking and applications-workshops, 2008 (AINAW 2008), Okinawa, Japan, 25–28 March 2008, pp 1672–1677
17. Huang T-K, Lee C-K, Chen L-J (2010) PRoPHET+: An adaptive PRoPHET-based routing protocol for opportunistic network. In: Proceedings of 24th IEEE international conference on advanced information networking and applications (AINA 2010), Perth, Australia, 20–13 April 2010, pp 112–119
18. Jain S, Fall K, Patra R (2004) Routing in a delay tolerant network. In: Proceedings of ACM SIGCOMM 2004, Portland/ Oregon/ USA, 30 Aug–3 Sept 2004, pp 145–158
19. Jain S, Shah RC, Brunette W, Borriello G, Roy S (2006) Exploiting mobility for energy efficient data collection in wireless sensor networks. In: ACM/Kluwer mobile networks and applications (MONET), vol 11, no 3, June 2006, pp 327–339
20. Kalman RE (1960) A new approach to linear filtering and prediction problems. Trans ASME J Basic Eng 82(Series D):35–45
21. Kathiravelu T, Ranasinghe N, Pears A (2010) A robust proactive routing protocol for intermittently connected opportunistic networks. In: Proceedings of seventh IEEE international conference on wireless and optical communications networks (WOCN 2010), Colombo, Sri Lanka, 6–8 Sept 2010, pp 1–6
22. Keranen A (2008) Opportunistic network environment simulator. Special assignment report, Helsinki University of Technology, Department of Communications and Networking, May 2008
23. LeBrun J, Chuah C-N, Ghosal D, Zhang M (2005) Knowledge based opportunistic forwarding in vehicular wireless ad hoc networks. In: Proceedings of 61st IEEE vehicular technology conference (VTC 2005-Spring), 30 May–1 June 2005, vol 4, pp 2289–2293
24. Leguay J, Friedman T, Conan V (2006) Evaluating mobility pattern space routing for DTNs. In: Proceedings of 25th IEEE international conference on computer communications (INFOCOM 2006), Barcelona, Spain, 23–29 April 2006, pp 1–10
25. Lilien L, Kamal ZH, Bhuse V, Gupta A (2006) Opportunistic networks: the concept and research challenges in privacy and security. In: Proceedings of NSF international workshop on research challenges in security and privacy for mobile and wireless networks (WSPWN 2006), Miami, March 2006, pp 134–147
26. Lin C-S, Chang W-S, Chen L-J, Chou C-F (2008) Performance study of routing schemes in delay tolerant networks. In: Proceedings of the 22nd international conference on advanced

information networking and applications—Workshops (AINAW '08), Washington, DC, USA 25–28 March 2008, pp 1702–1707

27. Lindgren A, Doria A, Schelen O (2003) Probabilistic routing in intermittently connected networks. ACM SIGMOBILE Mobile Comput Commun Rev 7(3):19–20
28. Liu Y, Niu J, Ma J (2009) Content encounter probability based message forwarding in opportunistic networks. In: Proceedings of Ist IEEE international conference on information science and engineering (ICISE '09), Nanjing, 26–28 Dec 2009, pp 2589–2594
29. Makhlouta J, Harkous H, Hutayt F, Artail H (2011) Adaptive fuzzy spray and wait: efficient routing for opportunistic networks. In: Proceedings of IEEE international conference on selected topics in mobile and wireless networking(iCOST), Shanghai, China, 10–12 Oct 2011, pp 64–69
30. Merugu S, Ammar M, Zegura E (2004) Routing in space and time in networks with predictable mobility. Technical report GIT-CC 04–7, Georgia Institute of Technology, 2004
31. Musolesi M, Hailes S, Mascolo C (2004) An ad hoc mobility model founded on social network theory. In: Proceedings of 7 th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems (MSWiM '04), Venice, Italy, 4–6 Oct 2004, pp 20–24
32. Musolesi M, Hailes S, Mascolo C (2005) Adaptive routing for intermittently connected mobile Ad Hoc network. In: Proceedings of sixth IEEE international symposium on world of wireless mobile and multimedia network (WOWMOM '05), Taormina-Giardini Naxos, Italy, 13–16 June 2005, pp 183–189
33. OMNeT++, Downloaded from http://www.omnetpp.org/
34. Pelusi L, Passarella A, Conti M (2006) Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. IEEE Commun Mag 44(11):134–141
35. Pelusi L, Passarella A, Conti M (2007) Encoding for efficient data distribution in multi-hop Ad hoc networks. In: Boukerche A (ed) Handbook of wireless Ad hoc and sensor networks. Wiley, New York
36. Perkins CE, Bhagwat P (1994) Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In: Proceedings of the ACM conference on communications architecture, protocols and applications (SIGCOMM '94), London, England UK, 31 Aug–02 Sept 1994, pp 234–244
37. Perkins CE, Royer EM (1999) Ad hoc on-demand distance vector routing. In: Proceedings of the 2nd IEEE workshop on mobile computing systems and applications, New Orleans, LA, Feb 1999, pp 90–100
38. Scott J, Gass R, Crowcroft J, Hui P, Diot C, Chaintreau A (2009) CRAWDAD data set cambridge/haggle (v. 2006–09-15), Downloaded from http://crawdad.cs.dartmouth.edu/cambridge/haggle, May 2009
39. Small T, Haas ZJ (2003) The shared wireless infostation model—a new Ad Hoc networking paradigm (or where there is a whale, there is a way). In: Proceedings of 4th ACM international symposium on mobile Ad Hoc networking and computing (MobiHoc 2003), Annapolis, MD, USA, 1–3 June 2003, pp 233–244
40. Spyropoulos T, Psounis K, Raghavendra CS (2004) Single-copy routing in intermittently connected mobile networks. In: Proceedings of first annual IEEE communications society conference on sensor and Ad Hoc communications and networks (SECON 2004), Santa Clara, CA, USA, 4–7 Oct 2004, pp 235–244
41. Spyropoulos T, Psounis K, Raghavendra CS (2005) Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In: Proceedings of ACM SIGCOMM workshop on delay-tolerant networking (WDTN '05), Philadelphia, PA, USA 22–26 Aug 2005, pp 252–259
42. Spyropoulos T, Psounis K, Raghavendra CS (2007) Spray and focus: efficient mobility-assisted routing for heterogeneous and correlated mobility. In: Proceedings of the fifth IEEE international conference on pervasive computing and communications workshops (PerComW '07), White Plains, NY, 19–23 March 2007, pp 79–85

43. The CMU Monarch Project‘s wireless and mobility extensions to ns-2. http://www.monarch.cs.rice.edu/cmu-ns.html
44. The Network Simulator (NS-2). www.isi.edu/nsnam/ns/
45. Vahdat A, Becker D (2000) Epidemic routing for partially connected ad hoc networks. Technical report CS-2000-06, Department of Computer Science, Duke University, Durham, NC, 2000
46. Widmer J, Le Boudec J-Y (2005) Network coding for efficient communication in extreme networks. In: Proceedings of ACM SIGCOMM workshop on delay-tolerant networking, Philadelphia, PA, 22–26 Aug 2005, pp 284–291
47. Woungang I, Denko MK (2011) Credit-based cooperation enforcement schemes tailored to opportunistic networks, Chap. 3. In: Denko M (ed) Mobile opportunistic networks: architectures, protocols and applications, Auerbach Publications/Taylor & Francis Group, Boca Raton/Florida. ISBN: 978-142-008-812-0, ISBN 10: 142-008-812-2, p 292
48. Yang J, Chen Y, Ammar M, Lee C (2005) Ferry replacement protocols in sparse MANET message ferrying systems. In: Proceedings of IEEE wireless communications and networking conference, New Orleans, USA 13–17 March 2005, pp 2038–2044
49. Zhang Z (2006) Routing in intermittently connected mobile Ad Hoc networks and delay tolerant networks: overview and challenges. IEEE Commun Surv Tutor 8(1):24–37
50. Zhao W, Ammar M, Zegura E (2004) A message ferrying approach for data delivery in sparse mobile Ad Hoc networks. In: Proceedings of 5th ACM international symposium mobile Ad Hoc networking and computing 2004 (MobiHoc ‘04), ACM Press, Tokyo, Japan, 24–26 May 2004, pp 187–198

Chapter 14
Incentive-Aware Opportunistic Network Routing

Greg Bigwood and Tristan Henderson

Abstract Opportunistic networking relies on cooperation between nodes to perform efficiently. Opportunistic routing protocols depend on nodes forwarding messages for each other, as otherwise the only delivery mechanism would be for the creator of a message to encounter the message destination node and deliver the message directly. Cooperative forwarding, however, incurs a cost to the forwarding nodes, both in terms of energy and storage, as nodes need to dedicate resources to forwarding. Due to these costs, nodes may wish to avoid the costs associated with participation in an opportunistic network, by not forwarding messages for other nodes. We discuss several attacks on opportunistic routing, and in particular, focus on selfishness of nodes. We discuss incentive mechanisms for opportunistic routing, before concluding with a discussion of some of the outstanding challenges in this area.

14.1 The Need for Incentives in Opportunistic Network Routing

An opportunistic network is a network in which there are nonstatic forwarding paths, frequent disconnections and where connections between nodes do not follow a predictable schedule. It may be the case that the source and destination of a message are never in the same network partition, or that they never simultaneously have their network connections enabled. In other words, an opportunistic network is a most challenged environment, where nodes are forced to *opportunistically* use any available communication mechanism in order to forward and route information.

G. Bigwood (✉) · T. Henderson
University of St Andrews,Fife, UK
e-mail: gjb4@st-andrews.ac.uk

T. Henderson
e-mail: tnhh@st-andrews.ac.uk

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3_14,

Opportunistic networks can be distinguished from other challenged networks, such as Delay-Tolerant Networks (DTNs), by the unpredictable and varied environment. For example, consider a DTN of Wi-Fi-enabled buses. The buses follow a predictable schedule. DTN routing protocols can therefore build a somewhat reliable encounter schedule and use this information to drive routing decisions. Contrast the Bus DTN with an opportunistic network of Bluetooth-enabled smartphones carried by commuters going about their daily lives. When these smartphones are in the range of one another, it may be possible for the devices to exchange messages. Encounters between commuters do not follow a schedule. Encounter patterns may emerge, but the lack of expected scheduled encounters necessitates a different approach to routing, and a different set of considerations as to what may affect the encounter patterns.

Opportunistic networking necessarily relies on cooperation between nodes, that is, the users participating in the network, to perform efficiently. Opportunistic routing protocols depend on nodes forwarding messages for each other, as otherwise the only delivery mechanism would be for the creator of a message to encounter the message destination node and deliver the message directly. Cooperative forwarding, however, incurs a cost to the forwarding nodes, both in terms of energy (battery power) and storage (the space required to store forwarded messages). Both of these are a constrained resource in mobile devices such as those used in opportunistic networks.

Due to the battery and storages costs of participating in an opportunistic network, nodes may wish to avoid these costs, by not forwarding messages for other nodes. We call this behaviour *selfishness*. Nodes behaving selfishly attempt to avoid costs to themselves, at the expense of the performance the other nodes receive from the network. When a node is behaving selfishly, it will not forward a message to another node at any time.

Figure 14.1 shows the results of an opportunistic network simulation where nodes act selfishly. In this trace-driven simulation, using the Reality Mining mobile phone dataset [7], nodes create and forward messages according to the Epidemic routing protocol [38]. We performed runs with differing numbers of selfish nodes. We see, that as the percentage of selfish nodes is increased, the performance of the network, i.e., the number of messages delivered, decreases rapidly as the percentage of selfish nodes increases. If we can detect and discourage selfish behaviour, it might be possible to achieve the same performance as if no nodes are selfish, even if all the nodes have a propensity for selfish behaviour.

How can we create the correct incentives for nodes to cooperate? Incentives, reputation and trust have been extensively studied in Wireless Mesh Networks [30], Peer-to-Peer Networks [26], Mobile Ad Hoc Networks [11] (MANETs) and more recently, in DTNs [16].

Huang et al. discuss the drawbacks of mobile DTN systems and state "without sufficient nodes cooperating to provide relaying functions a MANET cannot function properly" [11]. They go onto discuss the different drawbacks with various types of incentive systems for user co-operation. They find that in the early stage of a technology's development it is not necessary for the system to incentivise co-operation.

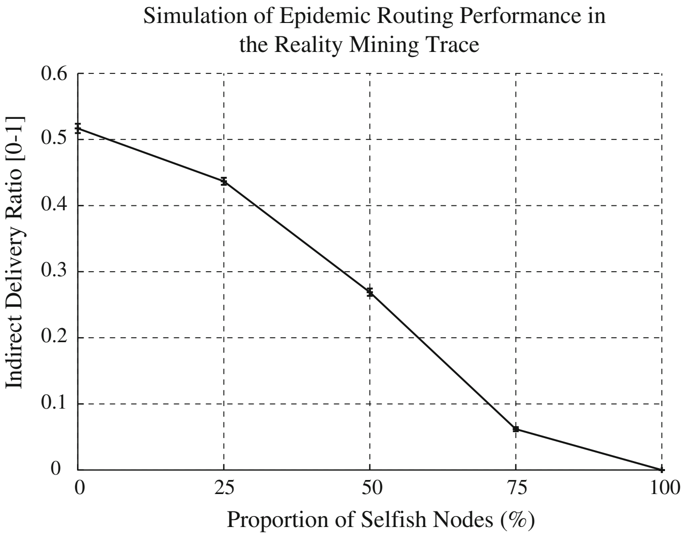


Fig. 14.1 Simulation of an opportunistic network application using Epidemic routing and the Reality Mining mobile phone dataset. Indirect delivery ratio is defined as the number of messages delivered that were not passed directly from the source to the destination, divided by the number of unique messages. As the proportion of selfish nodes in the network increases, network performance in terms of delivery ratio decreases. Selfish nodes do not pass messages that originate from nodes other than themselves. Error bars are included, but are too small to be seen

They do, however, state that this may be a problem in the future if the technology is to be used in a mainstream market. Thus, we look at attacks on opportunistic routing, incentives mechanisms and future challenges.

14.1.1 Attacks on Opportunistic Routing

We can view incentivising participation in the scope of malicious/selfish behaviour, as attacks and a lack of incentive for participation can both lead to reduced network performance. We now discuss attacks on opportunistic networks, as we cannot consider the improvements these networks can provide, without considering whether opportunistic networks are capable of providing those improvements in the face of a malicious attacker's behaviour.

14.1.1.1 Manipulation of Routes

Nodes may try and alter the speed at which they perform periodic control processes, for example increasing the route update frequency. This may affect the weighting of reports for example, or cause increased energy consumption [4].

Nodes may alter the intended path of a message by forwarding it to a node that was not on the intended path [4]. This can be an attempt to decrease the chance of a message arriving at the destination. This is also a way for an attacker to cause a node to have to do more than its fair share of forwarding, or reduce the demand on the malicious node.

14.1.1.2 Selective Maliciousness

Nodes may try to beat any reputation mechanism by only being selectively malicious. Either in the time dimension (by altering at which times they are malicious) or the node dimension (by only being malicious to certain nodes) [36].

By performing attacks on only a subset of the nodes (or for a short period of time), the attacker may avoid detection, while still gaining some return from its malicious behaviour.

14.1.1.3 Selfishness

If users behave selfishly, it harms the network as the nodes' messages may not get through. If all nodes behave selfishly, the only way to deliver messages would be via direct delivery. The users' economically rational desire to preserve their battery power affects this selfishness.

The vulnerability of opportunistic networks to selfishness is discussed in [14]. Karaliopoulos finds that DTNs are vulnerable to selfishness, and produces a metric for assessing the vulnerability, called the "deceleration factor". Panagakis et al. [27] also show that the opportunistic routing protocols can be susceptible to selfishness, in particular Epidemic [38] and Spray-and-Wait [34, 35] (both of which are highly cited protocols).

Keränen et al. find that once around 30 % of the nodes in the network are selfish, the performance of the routing protocols degrade [15]. Shevade et al. find that the performance of the network can drop to as low as 20 % if nodes are selfish [32]. Thus, we see that we should focus on incentivising user participation alongside protecting against attacks on the network, that a focus on the incentives for participation in opportunistic routing is needed.

Xu et al. [39] look at the different distributions of altruism for opportunistic sharing. They find that nodes with lots of contacts cannot maintain a uniform altruism to all connections due to resource constraints. They find that the "important" nodes should be favoured, as they are responsible for maintaining the short diameter of the network. This fits with the view of certain nodes in the network (hubs) having more influence over the transmission paths of messages than the average node. Solis et al. describe a mechanism that uses classification for the nodes to prioritise the traffic of nodes that contribute towards routing [33].

It may also be the case that nodes are selfish to the point where they do not forward messages on behalf of other nodes, but will pass messages to the destination if they meet them. This may be acceptable, depending on the routing protocol in use.

14.2 Incentivising Participation

We have seen in Sect. 14.1.1 that there are several attacks against opportunistic network routing. We now focus on incentivising nodes to participate in routing. Burgess et al., however [5], show that opportunistic networks are robust to many attacks, without even having to use authentication. They find that because the routing protocols frequently use a lot of message replication, opportunistic networks perform well in the face of attacks. We should, therefore consider selfishness as more of a threat than malicious attacks.

We also need to consider the incentives that users have to participate in a routing protocol at all. It would be rational for nodes to forward their own messages at all possible opportunities and to expect other nodes to forward all of their messages, whilst they themselves do not forward messages on behalf of any other nodes. While rational, this behaviour is selfish, to avoid costs, or to reduce congestion due to other nodes traffic, so that the selfish node's messages have an increased likelihood of reaching the destination. Sun et al. develop a trust measure for distributed networks [36], and argue that a trust measure is necessary to facilitate cooperation.

When Resta et al. look at node cooperation [29], they find that even a small number of nodes co-operating leads to significant improvements over the case where all nodes are selfish. If we can incentivise a small proportion of nodes to participate it could lead to increased routing performance. We need to decide, however, how to incentivise this participation.

To incentivise nodes to participate in forwarding, we can create an undesirable consequence for not following the routing protocol, or we can encourage participatory nodes through rewards. Opportunistic networks are mobile and ad hoc. The incentive mechanism thus needs to operate in a distributed fashion. For nodes to know whether or not a particular node is malicious or altruistic, we need to provide a common scale that nodes can be compared against. To do this, we can incorporate the concept of a computable reputation score for nodes, thus enabling the nodes to punish or reward nodes for their behaviour.

Many different mechanisms exist to create incentives to discourage selfishness [21]: from bartering (a direct exchange of services), to currency (behaviour that benefits other users earns measurable credits exchangeable for services). In the middle of this spectrum lies asynchronous bilateral trading: nodes perform actions to benefit one another, but not necessarily at the same point in time.

To combat selfishness, we must first detect it. Several approaches use "watchdog" mechanisms [1, 4, 10, 23–25], where a third node oversees a message exchange between two nodes to verify its authenticity. Such an approach, however, is inappropriate for opportunistic networks, as routes are rarely static and the inter-contact times are large; neighbours are not consistently available to monitor the behaviour of one another.

For a disconnected opportunistic network, it is therefore necessary to rely on the encounters between nodes as the only way to exchange data and incentive mechanism control traffic. The most common detection approach for opportunistic networks is for all nodes to monitor their own encounters, and to exchange their opinions of other nodes when they interact. Nodes then use these collated opinion data to make decisions about the trustworthiness of individual nodes. Liu and Issarny [20] describe a recommendation model, which is used to allow nodes to share their opinion of nodes so that malicious behaviour can be detected.

Liu et al. argue in favour of using reputation of recommendations as well as reputation of service provided by nodes [20]. They discuss how to force nodes to reveal the control information and reputation information.

Many mechanisms incentivise participation by encouraging the other nodes not to forward the selfish node's messages [1, 5, 10, 21, 22, 24, 25, 32]. Either selfish nodes do not have access to credits, or other nodes do not forward their messages because their reputation is below a certain threshold.

Yu et al. reputation system for peer-to-peer (P2P) networks has nodes build opinions of other nodes by analysing the quality of service (QoS) that they receive from these nodes [41]. They also use a measure of credibility for the nodes in the network. We refer to this protocol as YSS.

YSS includes a rating-discovery algorithm that maintains consistency of ratings across the network. P2P networks, however, have different properties to opportunistic networks. Even though there is potentially high churn in peer-to-peer networks, it is generally assumed that direct connectivity between any two nodes in the peer-to-peer network is possible, which is unlikely to be true in an opportunistic network. In opportunistic networks, it is harder for nodes to establish the opinion of nodes, as they cannot connect to arbitrary nodes to request reputation information.

To verify the reputation information, we must be able to prove that the reputation is based on experiences—how can we prove that messages were exchanged, or that encounters occurred? One way to validate encounters is to use encounter tickets [16], a cryptographic mechanism that nodes can use to prove encounters and message exchanges took place. When two nodes meet they create a data item containing the timestamp of the encounter, known as an encounter ticket. Assuming the nodes are using PKI, both nodes sign the ticket with their private keys. This provides non-repudiation for encounters. Encounter tickets allow nodes to build up a history of message exchanges to use to construct an opinion of other nodes. Nodes can exchange encounter tickets and opinions during encounters.

SMART is a credit-based multilayer incentive mechanism for DTN [42]. SMART uses provable exchanges and layered coins. This protocol requires the nodes to connect to a trusted authority for credit checks for nodes to send messages. Lu et al. also use an encounter-ticket-based incentive mechanism [22], but this requires a trusted authority (an out-of-band oracle), which makes it inappropriate for opportunistic networks. Li et al. propose RADON [17], which uses an interaction history-based approach, combined with watchdog nodes. RADON floods control messages to the network, potentially consuming lots of nodes' resources.

Any incentive protocol that relies on trust and reputation needs initial reputation values for the nodes [32]. Self-Reported Social Networks (SRSNs) provide information that is available before network startup. Social community information has been used to thwart node capture attacks [6], perhaps we can use SRSN information to bootstrap the incentive mechanism?

Taking advantage of existing social-network information may a help routing protocols and incentive mechanisms meet the demands users have for the ability to be selfish, while mitigating the damage this selfishness causes.

The SSAR protocol [19] treats opportunistic forwarding as a multiple knapsack problem. SSAR considers the demands of performance and selfishness, and prioritises packets from those nodes that have strong social ties to forwarding nodes. These social tie data are input by the user, and represent an SRSN.

The RELICS protocol [37] encourages cooperation through ranking. Nodes estimate the likelihood of message delivery for each of the nodes they encounter, and use this to rank nodes. A node's rank is improved by being on the forwarding path of successful delivery. Open message delivery a "delivery receipt" is generated, which is passed back to the sender of the message, to notify the sender of delivery. Nodes adjust their energy expenditure to meet a desired delivery ratio threshold (decided *a priori*). By expending more energy (forwarding more messages), nodes can hope to deliver more messages, increasing their rank with other nodes. Similarly, if their delivery ratio is above the threshold, nodes drop their energy expenditure. The RELICS protocol requires nodes to flood control messages to the network. It may be the case that not all of the nodes receive the control messages, and potentially there is a large energy overhead for control messages.

We can see in Table 14.1 several incentive mechanisms for incentive-aware routing, and the attack to which they are tailored to mitigate.

14.2.1 Attacks Against Incentive Mechanisms

Now that we have considered several attacks against routing mechanisms, and discussed incentive mechanisms, we consider attacks against incentive mechanisms.

Table 14.1 Incentive mechanisms: threats and incentives

Mechanism	Threat	Defence/Incentive
YSS [41]	Selfishness	Qos based credibility and ranking
RELICS [37]	Selfishness	Delivery receipts and limited energy expenditure
SMART [42]	Selfishness	Provable exchanges and layered coins
RADON [17]	Tailgating	Watchdog nodes and interaction prediction
MobiRate [28]	Defamation	Tamperevident tables
SSAR [19]	Selfishness	SRSN information

By considering the ways in which the incentive mechanisms are vulnerable, we can design incentive mechanism against these attacks.

14.2.1.1 Exploiting Friendship Mechanisms

Incentive mechanisms that use the concept of friendship or node pairing must take care not to incentivise nodes to accept all other nodes as friends or accept no nodes as friends. This can occur if the incentive mechanism hands out extra credits to nodes that are "friends" with the individual paying the credit charge, or perhaps by increasing the payout to individuals who are not friends with the node charged for sending the message.

14.2.1.2 Increasing Trust Through Epidemic Behaviour

It may be possible for a node to improve its score by ignoring the routing protocol and flooding messages on all network paths. This would increase the offending node's standing in other nodes' eyes as they are being helpful. If all nodes took this approach, however, the network would experience reduced delivery ratios due to flooding the network (a tragedy of the commons). This attack is still economically rational [19], as a node could decide to be selfish when it needs to be (when experiencing low energy), yet can gain significant trust/reputation by forwarding when it has a large amount of energy, e.g., when charging.

14.2.1.3 Tailgating

Nodes can follow another node, to create the appearance of a social connection or to collect records of encounters. In some systems, nodes use encounter records as a form of credit [5, 17, 18]. If the attacker can increase the amount of time that it spends with a node, then the other nodes in the network may give the attacker messages for the intended victim of tailgating. The attacker may then refuse to deliver these messages to the victim node, effectively cutting it off from the network. This attack can normally be solved outside of the incentive mechanism. The victim of the tailgating attack may be able to take legal action against the attacker for stalking.

14.2.1.4 Manipulation of Control Traffic

In this attack, nodes may refuse to submit their credits or not exchange control information [4, 9, 20, 42]. This behaviour makes it harder for the reputation mechanism to detect the selfishness, as there is less information to rely upon. Nodes may also try to offer routes that do not exist, by advertising that they are on the optimum forwarding path [4, 5, 41]. Many incentive mechanisms assign a high reputation value to a node that is doing a large share of forwarding. By reporting to be doing a higher share of forwarding than they are, the attacker can increase its reputation, as well as potentially avoiding its share of required message forwarding.

Nodes may create false acknowledgments of messages arriving at the destination [5]. This can be an attempt to avoid nodes asking the attacker to forward messages by causing other nodes to drop the messages that they may be holding.

Nodes may choose to create fake error messages, by claiming that a message was corrupt [4]. This would allow them to avoid having to forward any more copies of the message that they have been given.

14.2.1.5 Defamation

Nodes may choose to create false reputation claims about nodes, e.g., claiming that a node is malicious when it is not [12, 20, 22, 36, 41, 42]. This type of attack allows the attacker to potentially increase its own probability of nodes giving it a message in the future, and therefore gain credits. This would allow the attacker to intercept more messages. The MobiRate [28] ratings enforcement mechanism uses tamperevident tables and control traffic exchange to ensure ratings returned are accurate and finds reputable nodes.

14.2.1.6 Exploiting Detection Algorithms

In a credit-based system nodes must perform actions (altruistic behaviour) that they use to accrue credits, which they exchange for ability to forward messages [42]. If nodes can purchase more credits for the system, then they can afford to be more selfish: behaviour that does not benefit the network. If nodes all purchase more credits instead of following the routing protocol, then we are left with the same results as if there were no incentive mechanism at all. The cost of credits should therefore be set appropriately high. We must place limits on the number of credits that are needed to prevent credit explosion. Kangasharju et al. use this approach for opportunistic networks [13].

A further problem for credit-based mechanisms is that the system must not incentivise nodes to hoard credits. Especially if credits can be purchased, rather than earned through self-less actions.

If the incentive mechanism incentivises nodes to behave in a network it is possible that the nodes may start to behave selfishly again if they can avoid detection for a period of time. It may also be possible in mechanisms without credits, that nodes only stop being selfish while they are detected as being selfish, and then resort to being selfish again once they are no longer considered selfish. There may be a window in which nodes that are being selfish are not detected as such. We need to look at how well mechanisms detect the selfishness, as this detection-delay exploit may be quite large.

Table 14.2 Opportunistic routing and incentive mechanism threats and mitigations

Attack	Defence
Selfishness	An incentive mechanism
Traffic deviation	Watchdog nodes
Fake routes	Permit (within reason)
Fake error messages	Reputation mechanism
Increase route update frequency	Encounter tickets
Silent route change	Not applicable; no static routes
Collusion	Remove distributed mechanisms
Tailgating	No practical defence; solve out-of-band
Adding/removing nodes on path	PKI
Defamation	Encounter tickets
Submission refusal	Detect through reputation mechanism
Sybil attacks	Self-reported social networks
Trying to live in the grace period	Better detection mechanisms

Incentive mechanisms in which nodes gain reputation/trust by being on the delivery path, may encourage nodes to drop older messages. Nodes would be incentivised to drop messages that offer them a minimal chance of reputation improvement. This reduces network performance because less messages get through to the destination.

14.2.1.7 Collusion

Nodes may collude and work together to perform attacks [20, 41]. Rather than being an attack itself, this is a way for nodes to modify an attack strategy. Collusion adds another level of consideration for distributed reputation and incentive mechanisms designers to consider. For example, if nodes can pollute the reputation mechanism with false reputation reports, it may make accurately detecting malicious behaviour difficult.

14.2.1.8 Sybil Attacks

Malicious attackers may create several fake identities to mask their behaviour or alter the outcome of any reputation mechanism or routing decisions [36, 41]. This allows the attacker to use a new identity in the network if the reputation mechanism detects their previous malicious behaviour. In a sybil attack, the attacker uses multiple identities to increase the reputation of the attacker's main identity, in the same way as multiple nodes in collusion would.

We can see a summary of attacks on opportunistic routing, and incentive mechanisms in Table 14.2. Not all of the attacks discussed are a relevant threat to all opportunistic networks, as not all opportunistic networks rely on the same routing or incentive mechanisms. While many of the attacks can be prevented with adequate security protocols, however, it is important that the network can provide the intended service to users. If users are not incentivised to follow the routing protocols, nodes will not follow the routing protocols, thus, research into prevention of attacks against the routing protocols would be irrelevant. We believe, therefore, that incentivising participation and avoiding selfishness as the most important of the challenges facing opportunistic routing.

14.3 Incentive Mechanisms for Opportunistic Networks

In Sect. 14.2, we have seen the need for incentive mechanisms. Most of the incentive mechanisms discussed, were not specifically for opportunistic networks. Opportunistic networks, however, have some unique properties that we should consider.

First, incentive mechanisms must operate as soon as the network begins. Routing mechanisms may have warm up periods in which they are not effective, so we are faced with the challenge of how to ensure the incentive mechanisms works effectively.

Second, because opportunistic networks are designed to operate in geographical areas without infrastructure connectivity, we are unable to use out of band connections to infrastructure elements such as oracles. Centralised points of failure such as oracles and infrastructure connections also clash with the distributed architecture of opportunistic networks.

Third, there are power and networking constraints in opportunistic networks that are not found in other networks such as peer-to-peer networks. We must, therefore, conserve each node's battery and storage. Incentive mechanisms that require frequent control plane messages, for example delivery receipts routed back to the message source, may be inappropriate for opportunistic networks. All data, including control plane messages, must be forwarded opportunistically. This requirement may decrease the performance of MANET-based incentive mechanisms that rely on connected end-to-end paths for control plane messages.

Fourth, opportunistic networks may do not share a similar concept of Quality of Service with networks such as peer-to-peer networks. In a peer-to-peer network, for example, we may use upload speed and upload to download ratio as measures of QoS that a node provides to the network. In an opportunistic network, we cannot use these as measures.

One incentive mechanisms that aims to address these issues is IRONMAN [3]. IRONMAN uses pre-existing social network information to bootstrap trust relationships. By making use of existing information, from outside the network itself, IRONMAN does not require an oracle, infrastructure network, watchdog nodes nor flooded delivery receipts.

By viewing the members of a node's SRSN that are also members of the opportunistic network as more trustworthy, we can exploit this implicit trust relationships provided by the opportunistic network users. Detecting selfish behaviour quickly

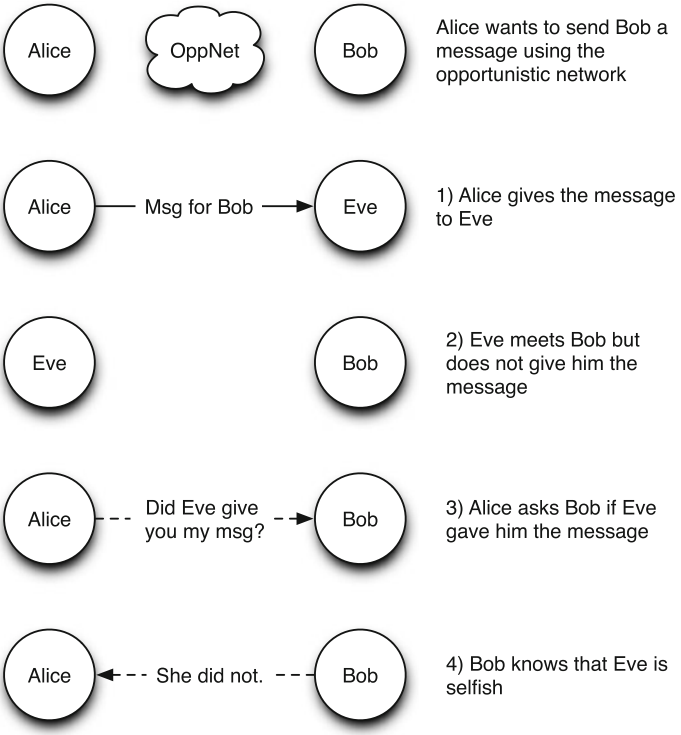


Fig. 14.2 Nodes keep a history of their encounters and message exchanges. When nodes meet they exchange histories to detect selfishness

reduces the amount of transmissions to (and due to) selfish nodes, which reduces energy wasted.

When using the IRONMAN protocol, nodes keep encounter histories, that they use to confirm delivery of a message when encountering the destination. We can see an example of encounter histories used for detection in Fig. 14.2.

To compare the performance of the incentive mechanisms, we must decide on appropriate metrics. We consider the negative impact of selfishness on the network using the following metrics:

1. *Detection Time*. The time that it takes a mechanism to correctly detect selfish behaviour in a node. This is the average time between a node choosing to behave selfishly, and the time that a node is detected as selfish. A mechanism with a low detection time will minimise the impact selfish behaviour has on the network.
2. *Detection Accuracy*. The proportion of selfish nodes that were correctly detected as selfish by a mechanism. An ideal mechanism will have a low Detection Time and a high Detection Accuracy.

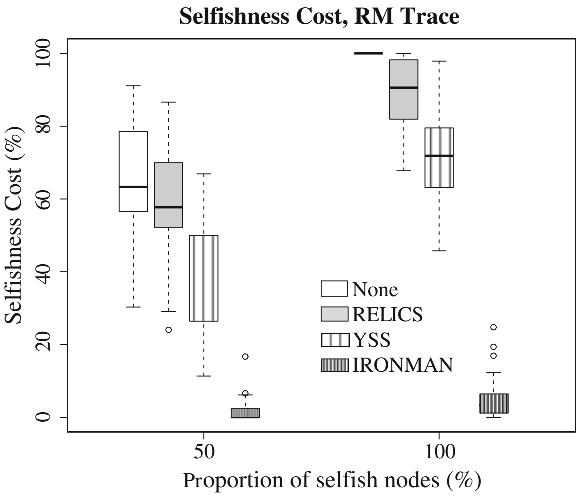


Fig. 14.3 A simulation of Epidemic routing over the reality Reality Mining trace. We compare the selfishness cost when two proportions of nodes behave selfishly. Nodes have a finite buffer, energy and message TTL. IRONMAN greatly outperforms the other mechanisms, with both YSS mechanisms performing similarly, followed by RELICS

3. *Selfishness Cost*. The proportion of forwarded messages (medium accesses) that were generated as a result of a node creating a message while it was selfish. In some respects this can be seen as the "goodput" of a network with selfish nodes; a mechanism with a low Selfishness Cost is effectively maximising the use of the network by cooperative nodes.

Using these metrics, we can compare the performance of several incentive mechanisms under simulation. In Fig. 14.3, we see the relative selfishness cost in a simulation of: no incentive mechanism, YSS, RELICS and IRONMAN. We see that all schemes outperform no incentive mechanism when all nodes are selfish, and that IRONMAN successfully incentives nodes to share information [3].

By bootstrapping the trust mechanism using SRSNs and using encounter histories to confirm delivery of messages, IRONMAN out-performs existing mechanisms without requiring an oracle or infrastructure network, nor delivery receipts. This shows that incentive mechanisms designed for opportunistic routing are useful, and motivates future work in this area.

14.4 Challenges for Incentive Aware Routing

We have discussed several effective incentive mechanisms for opportunistic networks. There are, however, many challenges and many avenues for future research in the area. We can group these challenges as follows:

14.4.1 User Behaviour

There are several questions related to user behaviour related to incentives to participate in opportunistic routing. In particular, how do we handle malicious behaviour of certain nodes? We must be able to incentivise against selfishness, while also protecting the network against malicious behaviour.

It may be the case that nodes behave altruistically, and in accordance with the routing protocol in use, but in certain circumstances still behave selfishly. One example may be when a user's battery is running low, they do not route messages for other nodes. Incentive mechanisms need to consider human factors such as this.

How can nodes corroborate information, especially with regards to message timings? Because opportunistic network nodes work in a disconnected environment, converging on accurate message timestamps is a difficult challenge. Creating protocols that rely on alternative methods of verifying meetings and interactions, such as encounter tickets, may be provide an alternative to reliable, accurate event timings.

14.4.2 Using Social Network Information

We have seen that using social network information can improve incentive mechanisms for opportunist routing. Perhaps, we can classify users based on their social network information [2]? These classifications may help in routing and incentive mechanisms.

As opportunistic routing patterns may be similar to human social network patterns, opportunistic network researchers may find that some of the issues that arrive in this area have been addressed outside of the field of networking. In particular, the field of social-science and sociology could provide some useful new partnerships for cross-discipline work.

14.4.3 Cross-Layer Information Use

Many opportunistic network applications might themselves involve social networks, for instance, mobile social networks, crowdsourcing, or participatory sensing. Might it be useful to expose trust relationships from the routing layer to the application layer, or vice versa?

Perhaps, we could use application layer detection of misbehaving nodes, such as anomalous crowdsourced data, to inform routing decisions? Such further study will require both routing protocol development and application deployment.

14.4.4 Modelling Social Network Behaviour

Several of the incentive mechanisms discussed assume that members of the same social network will be more likely to trust each other. But if behaviour is contagious across a social network, as proposed by Fowler and Christakis [8], then perhaps selfish behaviour might also propagate, leading to new incentive challenges. Predicting user location [31] and interaction is a hot topic for both opportunistic and pervasive networks.

14.4.5 Academic Challenges

Conducting research in the area of Opportunistic Networks frequently involves the collection and analysis of large datasets that potentially contains participants sensitive personal data. Such data can include locations visited, social network and interaction patterns and phone call logs.

We must take care when analysing and distributing these data that we treat this personal information ethically and in accordance with applicable laws and data protection guidelines.

Collecting a dataset of a large size can, therefore, be a daunting, expensive and difficult procedure. Researchers can, and should, share data with the community, thus enabling corroboration of results and future work in the area. Sites such as the wireless data archive CRAWDAD [40] try to address this problem by providing a data repository for data traces with links to the associated papers. Currently, a lack of availability of a data trace from a large-scale deployment means that thorough evaluation of opportunistic networking remains a challenge.

14.4.6 Metrics for Analysing Incentive Mechanisms

There does not seem to be consensus on how best to compare and analyse the incentive mechanisms for opportunistic networks. Simulations work well for analysing encounter patterns, but incentive mechanisms may cause users to alter their behaviour, making these simulations complex. There is currently no standard simulator to use for incentive mechanisms and opportunistic routing.

User data traces would go some way to address this problem, but user behaviour would likely be specific to the analysed protocols. At the moment, however, there is currently no large-scale opportunistic network in deployment (neither in the academic nor commercial sectors).

Furthermore, what does a “fair” forwarding distribution look like? Should users be allowed some degree of selfishness? Answers to these questions will largely be deployment specific. Perhaps, opportunistic routing algorithms and the associated incentive mechanisms need to be able to be tuned for each deployment? How would an opportunistic network with heterogeneous routing protocols and incentive mechanisms work?

14.5 Conclusion

Incentives mechanisms for participating in opportunistic routing will be vital for any opportunistic networking deployment where users bare the costs of routing messages. Without an incentive mechanism, rational user selfishness will greatly reduce network throughput.

We have seen that requirements for incentive mechanisms in opportunistic networks differ from seemingly similar mobile networks such as MANETs and DTNs. Several recent incentive mechanisms designed for opportunistic networks focus on utilising data gathered from the users about their encounter patterns, so called self-reported social networks.

The requirements of opportunistic networks of incentive mechanisms leads to a set of unique challenges, many which have not been investigated. Solving these problems may lead to a greater understanding of opportunistic networks in general, and to interesting cross-discipline advances in the fields of sociology and psychology.

References

1. Anantvalee T, Wu J (2007) Reputation-based system for encouraging the cooperation of nodes in mobile ad hoc networks. In: Proceedings of the IEEE international conference on communications (ICC ’07), June 2007, pp 3383–3388. doi:10.1109/ICC.2007.560
2. Bigwood G, Henderson T, (2010) Social roles for opportunistic forwarding. In: Proceedings of the 2nd international workshop on mobile opportunistic networking (MobiOpp ’10), February 2010. ACM, New York, pp 199–200. doi:10.1145/1755743.1755787
3. Bigwood G, Henderson T (2011) IRONMAN: using social networks to add incentives and reputation to opportunistic networks. In: Proceedings of the IEEE 3rd international conference on social computing (SocialCom), IEEE, October 2011, pp 65–72. doi:10.1109/PASSAT/SocialCom.2011.60
4. Buchegger S, Le Boudec JY, (2002) Performance analysis of the CONFIDANT protocol. In: Proceedings of the 3rd ACM international symposium on mobile ad hoc networking & computing (MobiHoc ’02), 2002. ACM, New York, pp 226–236. doi:10.1145/513800.513828
5. Burgess J, Bissias GD, Corner MD, Levine BN, (2007) Surviving attacks on disruption-tolerant networks without authentication. In: Proceedings of the 8th ACM international symposium on mobile ad hoc networking and computing (MobiHoc ’07), 2007. ACM, New York, pp 61–70. doi:10.1145/1288107.1288116
6. Conti M, Di Pietro R, Gabrielli A, Mancini LV, Mei A, (2010) The smallville effect: social ties make mobile networks more secure against node capture attack. In: Proceedings of the 8th ACM international workshop on mobility management and wireless access (MobiWac ’10), 2010. ACM, New York, pp 99–106. doi:10.1145/1868497.1868514
7. Eagle N, Pentland AS (2005) CRAWDAD data set mit/reality (v. 2005–07-01). http://crawdad.org/mit/reality. Accessed July 2005

8. Fowler JH, Christakis NA (2010) Cooperative behavior cascades in human social networks. Proc Natl Acad Sci 107(12):5334–5338. doi:10.1073/pnas.0913149107
9. Ganeriwal S, Balzano LK, Srivastava MB (June 2008) Reputation-based framework for high integrity sensor networks. ACM Trans Sens Netw 4(3):1–37. doi:10.1145/1362542.1362546
10. He Q, Wu D, Khosla P (2004) SORI: a secure and objective reputation-based incentive scheme for ad-hoc networks. In: Proceeding of the wireless communications and networking conference (WCNC), 2004, vol 2, pp 825–830. doi:10.1109/WCNC.2004.1311293
11. Huang E, Crowcroft J, Wassell I, Rethinking incentives for mobile ad hoc networks. In: PINS '04: Proceedings of the ACM SIGCOMM workshop on practice and theory of incentives in networked systems, 2004. ACM, New York, pp 191–196. doi:10.1145/1016527.1016532
12. Huang KL, Kanhere SS, Hu W, (2010) Are you contributing trustworthy data? The case for a reputation system in participatory sensing. In: Proceedings of the 13th ACM international conference on modeling, analysis, and simulation of wireless and mobile systems (MSWIM '10), 2010. ACM, New York, pp 14–22. doi:10.1145/1868521.1868526
13. Kangasharju J, Heinemann A, (2008) Incentives for opportunistic networks. In: 22nd international conference on advanced information networking and applications: Workshops (aina workshops 2008) March 2008. IEEE, Los Alamitos, pp 1684–1689. doi:10.1109/WAINA.2008.90
14. Karaliopoulos M (2009) Assessing the vulnerability of DTN data relaying schemes to node selfishness. IEEE Commun Lett 13(12):923–925. doi:10.1109/LCOMM.2009.12.091520
15. Keränen A, Pitkänen M, Vuori M, Ott J (2011) Effect of non-cooperative nodes in mobile DTNs. In: Proceedings of the 12th IEEE international symposium on a world of wireless, mobile and multimedia networks (WoWMoM '11), June 2011. doi:10.1109/WoWMoM.2011.5986144
16. Li F, Wu J, Srinivasan A (2009) Thwarting blackhole attacks in disruption-tolerant networks using encounter tickets. In: IEEE INFOCOM 2009: the 28th conference on computer communications, IEEE, April 2009, pp 2428–2436. doi:10.1109/INFCOM.2009.5062170
17. Li N, Das SK, (2010) RADON: reputation-assisted data forwarding in opportunistic networks. In: Proceedings of the second international workshop on mobile opportunistic networking (MobiOpp '10), February 2010. ACM, New York, pp 8–14. doi:10.1145/1755743.1755746
18. Li N, Das SK (2011) A trust-based framework for data forwarding in opportunistic networks. Ad Hoc Netw. doi:10.1016/j.adhoc.2011.01.018
19. Li Q, Gao W, Zhu S, Cao G (2012) A routing protocol for socially selfish delay tolerant networks. Ad Hoc Netw 10(8):1619–1632. doi:10.1016/j.adhoc.2011.07.007 Special issue on social-based routing in mobile and delay-tolerant networks
20. Liu J, Issarny V (2004) Enhanced reputation mechanism for mobile ad hoc networks. In: Jensen C, Poslad S, Dimitrakos T (eds), Trust management. Lecture notes in computer science, vol 2995, chapter 5. Springer, Berlin, pp 48–62. doi:10.1007/978-3-540-24747-0_5
21. Liu Z, Hu H, Liu Y, Ross KW, Wang Y, Mobius M, (2010) P2P trading in social networks: the value of staying connected. In: Proceedings of the 29th conference on information communications (INFOCOM '10), 2010. IEEE Press, Piscataway, pp 2489–2497. doi:10.1109/INFCOM.2010.5462077
22. Lu R, Lin X, Zhu H, Shen X, Preiss B (2010) Pi: A practical incentive protocol for delay tolerant networks. IEEE Trans Wireless Commun 9(4):1483–1493. doi:10.1109/TWC.2010.04.090557
23. Mahajan R, Rodrig M, Wetherall D, Zahorjan J (2005) Sustaining cooperation in multi-hop wireless networks. In: Proceedings of the 2nd conference on symposium on networked systems design & implementation (NSDI'05), 2005, vol 2. USENIX Association, Berkeley, pp 231–244. http://portal.acm.org/citation.cfm?id=1251220
24. Marti S, Giuli TJ, Lai K, Baker M, (2000) Mitigating routing misbehavior in mobile ad hoc networks. In: Proceedings of the 6th annual international conference on mobile computing and networking (MobiCom '00), 2000. ACM, New York, pp 255–265. doi:10.1145/345910.345955253C

25. Michiardi P, Molva R (2002) Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In: Proceedings of the IFIP TC6/TC11 sixth joint working conference on communications and multimedia security, 2002. Kluwer, Deventer, pp 107–121. http://portal.acm.org/citation.cfm?id=647802.737297
26. Palomar E, Alcaide A, Ribagorda A, Zhang Y (2012) The peer's dilemma: a general framework to examine cooperation in pure peer-to-peer systems. Comput Netw 56(17):3756–3766. doi:10.1016/j.comnet.2012.08.012
27. Panagakis A, Vaios A, Stavrakakis L (2007) On the effects of cooperation in DTNs. In: Proceedings of the 2nd international conference on communication systems software and middleware (COMSWARE 2007), IEEE, January 2007, pp 1–6. doi:10.1109/COMSWA.2007.382571
28. Quercia D, Hailes S, Capra L, (2008) Mobirate: making mobile raters stick to their word. In: Proceedings of the 10th international conference on ubiquitous computing (UbiComp '08), 2008. ACM, New York, pp 212–221. doi:10.1145/1409635.1409664
29. Resta G, Santi P, (2009) The effects of node cooperation level on routing performance in delay tolerant networks. In: Proceedings of the 6th annual IEEE communications society conference on sensor, mesh and ad hoc communications and networks (SECON'09), 2009. IEEE Press, Piscataway, pp 413–421. doi:10.1109/SAHCN.2009.5168977
30. Saxena N, Denko M, Banerji D (2011) A hierarchical architecture for detecting selfish behaviour in community wireless mesh networks. Comput Commun 34(4):548–555. doi:10.1016/j.comcom.2010.04.040
31. Scellato S, Musolesi M, Mascolo C, Latora V, Campbell A, (2011) NextPlace: a spatio-temporal prediction framework for pervasive systems. In: Lyons K, Hightower J, Huang E (eds) Proceedings of the 9th international conference on pervasive computing, June 2011. Springer, Berlin, pp 152–169. doi:10.1007/978-3-642-21726-510
32. Shevade U, Song HH, Qiu L, Zhang Y (2008) Incentive-aware routing in DTNs. In: 2008 IEEE international conference on network protocols, IEEE, October 2008, pp 238–247. doi:10.1109/ICNP.2008.4697042
33. Solis J, Asokan N, Kostiainen K, Ginzboorg P, Ott J (2010) Controlling resource hogs in mobile delay-tolerant networks. Comput Commun 33(1):2–10. doi:10.1016/j.comcom.2009.07.019
34. Spyropoulos T, Psounis K, Raghavendra CS, (2005) 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. ACM, New York, pp 252–259. doi:10.1145/1080139.1080143
35. Spyropoulos T, Psounis K, Raghavendra CS (2008) Efficient routing in intermittently connected mobile networks: the multiple-copy case. IEEE/ACM Trans Netw 16(1):77–90. doi:10.1109/TNET.2007.897964
36. Sun YL, Han Z, Yu W, Ray Liu KJ (2006) A trust evaluation framework in distributed networks: vulnerability analysis and defense against attacks. In: IEEE INFOCOM, 2006, pp. 230–236. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.84.5322
37. Uddin MY, Godfrey B, Abdelzaher T (2010) RELICS: In-network realization of incentives to combat selfishness in DTNs. In: Proceeding of the 18th IEEE international conference on network protocols (ICNP), October 2010, pp 203–212. doi:10.1109/ICNP.2010.5762769
38. Vahdat A, Becker D (2000) Epidemic routing for partially-connected ad hoc networks. Technical report, Duke University. http://issg.cs.duke.edu/epidemic/epidemic.pdf. Accessed April 2000
39. Xu K, Hui P, Li VO-K, Crowcroft J, Latora V, Lio P, (2009) Impact of altruism on opportunistic communications. In: Proceedings of the first international conference on ubiquitous and future networks (ICUFN'09), 2009. IEEE Press, Piscataway, pp 153–158. doi:10.1109/ICUFN.2009.5174303
40. Yeo J, Kotz D, Henderson T (2006) CRAWDAD: a community resource for archiving wireless data at dartmouth. Comput Commun Review 36(2):21–22

41. Yu B, Singh MP, Sycara K (2004) Developing trust in large-scale peer-to-peer systems. In: 2004 IEEE 1st symposium on multi-agent security and survivabilty, December 2004, pp 1–10. doi:10.1109/MASSUR.2004.1368412
42. Zhu H, Lin X, Lu R, Fan Y, Shen X (2009) SMART: a secure multilayer credit-based incentive scheme for delay-tolerant networks. IEEE Trans Veh Technol 58(8):4628–4639. doi:10.1109/TVT.2009.2020105

Editors' Biography

Dr. Isaac Woungang received his M.S. and Ph.D. degrees, all in Mathematics, from the Université de la Méditerranée-Aix Marseille II, France, and Université du Sud, Toulon & Var, France, in 1990 and 1994, respectively. In 1999, he received a M.S degree from the INRS-Materials and Telecommunications, University of Quebec, Montreal, Canada. From 1999 to 2002, he worked as a Software Engineer at Nortel Networks. Since 2002, he has been with Ryerson University, where he is now an Associate Professor of Computer Science and Coordinator of the Distributed Applications and Broadband (DABNEL) Lab at Ryerson University. During his sabbatical in 2008–2009, he was a Visiting Professor at National Ilan University, Taiwan. His current research interests include network security, computer communication networks, and mobile communication systems. Dr. Woungang has published 7 book chapters and over 75 refereed technical articles in scholarly international journals and proceedings of international conferences.

Dr. Woungang serves as Editor-in-Chief of the International Journal of Communication Networks and Distributed Systems (IJCNDS), Inderscience, U.K., and the International Journal of Information and Coding Theory (IJICoT), Inderscience, U.K, as Associate Editor of the International Journal of Communication Systems (IJCS), Wiley, the Computers & Electrical Engineering (C&EE), An International Journal, Elsevier, as Associate Editor of the International Journal of Wireless Networks & Broadcasting Technologies (IGI Global), as Associate Editor of the Journal of Convergence (FTRA, http://ftrai.org/joc/index.htm), and as Associate Editor of the Human-centric Computing and Information Sciences, Springer. He also served as Guest Editor for several special issues with several journals such as IET Information Security, International Journal of Internet Protocol Technology (Inderscience); Computer and Electrical Engineering (Elsevier), Mathematical and Computer Modelling (Elsevier), Computer Communications (Elsevier); The Journal of Supercomputing (Springer), Telecommunication Systems (Springer), and International Journal of Communication Systems (Elsevier).

Dr. Woungang edited six books in the areas of wireless ad hoc networks, wireless sensor networks, wireless mesh networks, pervasive computing, and coding theory, published by reputed publishers such as Springer, Elsevier, Wiley, and World Scientific. He is currently working on two more books. Also, he has been serving as

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3, © Springer Science+Business Media New York 2013

Symposium Co-Chair of the 29th IEEE Symposium on Reliable Distributed Systems (SRDS 2010), Organizing Chair of the 2nd, 3rd, and 4th International Workshop on Dependable Network Computing and Mobile Systems (DNCMS), Track Chair, and Program Chair of several International conferences; and in the Program Committees of over a dozen International Conferences. Since January 2012, Dr. Woungang serves as Chair of Computer Chapter, IEEE Toronto Section.

Dr. Sanjay K. Dhurandher received the M.Tech. and Ph.D. degrees in Computer Sciences from the Jawaharlal Nehru University, New Delhi, India. He is presently an Associate Professor and Head of the Advanced Centre CAITFS, Division of Information Technology, Netaji Subhas Institute of Technology (NSIT), University of Delhi, India. From 1995 to 2000, he worked as a Scientist/Engineer at the Institute for Plasma Research, Gujarat, India, which is under the Department of Atomic Energy, India. His current research interests include wireless ad hoc networks, sensor networks, computer networks, network security, and underwater sensor networks. He currently serves as Associate Editor for International Journal of Communication Systems, Wiley.

Prof. Alagan Anpalagan received the B.A.Sc. (H), M.A.Sc. and Ph.D. degrees in Electrical Engineering from the University of Toronto, Canada. He joined the Department of Electrical and Computer Engineering at Ryerson University in 2001 and was promoted to Full Professor in 2010. He served the department as Graduate Program Director (2004–2009) and the Interim Electrical Engineering Program Director (2009–2010). During his sabbatical in 2010–2011, he was a Visiting Professor at Asian Institute of Technology and Visiting Researcher at Kyoto University. Dr. Anpalagan's industrial experience includes working at Bell Mobility on 1xRTT system deployment studies (2001), at Nortel Networks on SECORE R&D projects (1997), and at IBM Canada as IIP Intern (1994).

Dr. Anpalagan directs a research group working on radio resource management (RRM) and radio access and networking (RAN) areas within the WINCORE Lab. His current research interests include cognitive radio resource allocation and management, wireless cross layer design and optimization, collaborative communication, green communications technologies, and QoE-aware femtocells. Dr. Anpalagan serves as Associate Editor for the IEEE Communications Surveys & Tutorials (2012-), IEEE Communications Letters (2010-), and Springer Wireless Personal Communications (2009-), and past Editor for EURASIP Journal of Wireless Communications and Networking (2004–2009). He also served as EURASIP Guest Editor for two special issues in Radio Resource Management in 3G+ Systems (2006) and Fairness in Radio Resource Management for Wireless Networks (2008). Dr. Anpalagan served as TPC Co-Chair of: IEEE WPMC'12 Wireless Networks, IEEE PIMRC'11 Cognitive Radio and Spectrum Management, IEEE IWCMC'11 Workshop on Cooperative and Cognitive Networks, IEEE CCECE'04/08 and WirelessCom'05 Symposium on Radio Resource Management.

Dr. Anpalagan served as IEEE Toronto Section Chair (2006–2007), ComSoc Toronto Chapter Chair (2004–2005), and Chair of IEEE Canada Professional Activities Committee (2009–2011). He is the recipient of the Dean's Teaching Award (2011), Faculty Scholastic, Research and Creativity Award (2010), and Faculty Service Award (2010) at Ryerson University. Dr. Anpalagan also completed a course on Project Management for Scientist and Engineers at the University of Oxford CPD Center. He is a registered Professional Engineer in the province of Ontario, Canada.

Prof. Athanasios V. Vasilakos is currently Professor at the Department of Computer and Telecommunications Engineering, University of Western Macedonia, Greece and Visiting Professor at the Graduate Programme of the Department of Electrical and Computer Engineering, National Technical University of Athens (NTUA), Greece. His research interests include computer networks, mobile computing, wireless communications, game theory, AI, bioinformatics, and digital arts.
Professor Vasilakos is Co-author (with W. Pedrycz) of the book entitled "Computational Intelligence in Telecommunications Networks", CRC Press, USA, 2001; the book entitled "Ambient Intelligence, Wireless Networking, Ubiquitous Computing", Artech House, USA, 2006; Co-author (with M. Parashar, S. Karnouskos, and W. Pedrycz) of the book entitled "Autonomic Communications", Springer, 2009; of the book entitled "Digital Arts" (in Greek); and Co-author (with Yan Zhang, Thrasyvoulos Spyropoulos) of the book entitled "Delay Tolerant Networks: Protocols and Applications", CRC Press, 2010.

Professor Vasilakos has published more than 200 publications in top international journals such as IEEE/ACM Transactions on Networking, IEEE T-Information Theory, IEEE JSAC, IEEE Transactions on Communications, IEEE Transactions on Wireless Communications, IEEE Transactions on Mobile Computing, ACM Transactions on Autonomous and Adaptive Systems, IEEE Transactions on Neural Networks, IEEE Transactions on Systems, Man. & Cybernetics, IEEE T-ITB, etc. and conferences in the area of Computer Networks, Mobile Computing, Wireless Networking, Game Theory, Evolutionary Game Theory, Bioinformatics, Wireless Healthcare, and Digital and Internet Arts.

Professor Vasilakos is chairing or has chaired several conferences such as BodyNets 2010, BIONETICS 2010, ACM IWCMC'09, and ICST/ACM Autonomics 2009 and he served or is serving in the TPC of several conferences/symposia including IEEE INFOCOM 2011, 2009, IEEE ICAC'08, 09, ICC'08, 09, 10, and 11, IEEE GLOBECOM 2009 and 2010, IEEE BIBE'08, IEEE/WIC/ACM Web Intelligence 2007 (WI'07), WEBIST'2007, IEEE WCCI'08, IEEE IJCNN'08, AAIA'07, IEEE CEC'2007/2008, IEEE FOCI'07, IEEE CIBCB'07, IEEE BIBE'07, IEEE INFOCOM(2001), 1st IEEE International Workshop on Specialized Ad Hoc Networks and Systems (SAHNS 2007), IEEE International

Conference on Mobile Ad hoc and Sensor Systems (MASS 2006 and MASS 2007), GC'07ASNS (IEEE GLOBECOM 2007, Ad hoc and Sensor Networking Symposium), 4th International Conference on Ubiquitous Intelligence and Computing (UIC-07), 2007.

Professor Vasilakos is the Editor-in-chief of the following Inderscience Publishers journals: International Journal of Adaptive and Autonomous Communications Systems (IJAACS, http://www.inderscience.com/ijaacs), International Journal of Arts and Technology (IJART, http://www.inderscience.com/ijart).

Professor Vasilakos has been or he is in the Editorial Board of several international journals: IEEE Communications Magazine (1999–2002 and 2008-), IEEE Transactions on Systems, Man. and Cybernetics (SMC, Part B, 2007-), IEEE Transactions on Information Theory in Biomedicine (TITB, 2008-), IEEE Transactions on Wireless Communications, ACM/Springer Wireless Networks, Wireless Communications and Mobile Computing (Wiley), EURASIP Journal on Wireless and Communications Networks (WCN), Computer Communications (Elsevier, 1988–1991), ACM Applied Computing Reviews (ACM ACR), Soft Computing (Springer), Information Sciences (Elsevier), International Journal on Computational Intelligence Research, International Journal on Cognitive Informatics and Natural Intelligence (IJCiNi), International Journal of Ad Hoc and Ubiquitous Computing (IJAHUC), ACM Computers in Entertainment (ACM CIE), Journal of Computational Intelligence in Bioinformatics and Systems Biology (JCIBSB), International Journal of Functional Informatics and Personalised Medicine (IJFIPM), International Journal of Mobile Communications (IJMC), International Journal of Computational Science (IJCS), International Journal of Internet Protocol Technology (IJIPT), Cluster Computing (Springer), Security and Communications journal (Wiley), Journal of Sensors (Hindawi), Journal of Supercomputing (Springer), and Telecommunications Journal (Springer).

Professor Vasilakos serves or has served as Guest Editor of several special issues in various: IEEE T-Systems, Man and Cybernetics (special issue on Computational Intelligence in Telecommunications Networks, 2003), ACM Transactions on Autonomous and Adaptive Systems, http://www.acm.org/pubs/taas/ (special issue on Autonomic Communications, Dec 2009; special issue on Ambient Intelligence, 2010; special issue on Foraging-based Optimization Algorithms, 2010); IEEE Journal on Selected Areas in Communications (JSAC) (special issues on Wireless and Pervasive Networks in Healthcare, May 2009; special issue on Vehicular Communications and Networking, 2010), Journal on Interactive Learning Research (special issue on Computational Intelligence in Web-based Education, Vol. 15, No. 4, 2004), IJCiNi (special issue on Ambient Intelligence (AmI) and Arts), IJCS, http://www.gip.hk/ijcs/ (special issue on Computational Arts), Journal of Computational Intelligence in Bioinformatics and Systems Biology (special issue on Classify the Classifiers: Investigating the Optimum Classification

Technique per case in Bioinformatics), IEEE T-Systems, Man, and Cybernetics Part B (special issue on Game Theory), IEEE Transactions on Information Technology in Biomedicine (special issue on Affective and Pervasive Computing in Biomedicine), IEEE T-Systems, Man, and Cybernetics Part B (special issue on Sensor and Actuator networks in cyber/physical systems), etc.

Professor Vasilakos is Chairman of the Telecommunications Task Force of the Intelligent Systems Applications Technical Committee (ISATC) of the IEEE Computational Intelligence Society (CIS).

Index

I. Woungang et al. (eds.), *Routing in Opportunistic Networks*,
DOI: 10.1007/978-1-4614-3514-3,