The Hong Kong Polytechnic University
Department of Computing

Routing Strategies and Cooperation Incentive Mechanisms for Delay Tolerant Networks

by

Honglong CHEN

A thesis submitted in partial fulfillment of the requirements for
the degree of Doctor of Philosophy
April, 2012

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

________________________(Signature)

Honglong Chen   (Name of Student)

ABSTRACT

In most of the conventional routing protocols, the messages are designed to be delivered along the always-connected end-to-end path between the source and destination. However, this kind of routing protocols are not applicable in delay tolerant networks (DTNs), in which the nodes are intermittently connected. DTNs, as an emergent communication paradigm, enable the transfer of messages among the intermittently connected nodes. Applications of DTNs include social networks, vehicular networks, pocket switched networks and habitat monitoring sensor networks, etc.

In DTNs, the nodes are generally with some characteristics, which should be carefully considered in the research. Firstly, they are always intermittently connected due to their high mobility, sparse deployment and short radio range. Secondly, they are always some resource-limited (e.g., bandwidth, buffer space, energy power, etc.) mobile devices such as smart phones or PDAs. Thirdly, they are always some distributed autonomous entities, which make the routing decision locally and independently. Such kind of characteristics will challenge the research of DTNs from the aspects of routing, cooperation incentive, security and privacy, etc.

Firstly, this dissertation proposes to use the contact expectation for the DTN routing. An expected encounter based routing protocol (EER) is proposed, which distributes multiple replicas of a message proportionally between two encounters according to their expected encounter values. In case of single replica of a message, EER makes the routing decision by comparing the minimum expected meeting delay to the destination. A community based routing protocol (CR) is further proposed, which takes advantages of the high contact frequency property of the community. The simulations demonstrate the effectiveness of the proposed EER and CR protocols under different network parameters.

Secondly, this dissertation proposes a cooperative routing protocol using the group feature for the DTNs under resource constraints, which includes a cooperative message transfer scheme and a buffer management strategy. In the cooperative message transfer scheme, the limited bandwidth is considered and the message transfer priorities are designed to maximize the delivery probability. In the buffer management strategy, by considering the constraint of buffer space, the cooperative message caching scheme is proposed and the dropping order of the messages is designed to minimize the reduced delivery probability. Finally, the simulations are conducted to demonstrate the effectiveness of the proposed group aware routing protocol under different network parameters.

Thirdly, this dissertation considers the noncooperative DTNs, in which the nodes may be selfish and reluctant to cooperate with each other in the message forwarding. Two credit-based rewarding schemes, called earliest path singular rewarding scheme (EPSR) and earliest path cumulative rewarding scheme (EPCR) respectively, are proposed to ensure the nodes truthfully forward the messages. The proposed rewarding schemes are proved to be incentive compatible. It is also proved that the payment for each delivered message in these schemes is upper bounded. Furthermore, the proposed rewarding schemes can prevent selfish nodes from having malicious behaviors. The real trace based simulations are conducted to illustrate the effectiveness of the proposed rewarding schemes.

Finally, this dissertation proposes to protect the end-to-end location privacy in the delay tolerant event collection systems, which are one of the typical DTN applications. Previous research only focuses on the location privacy of the source or sink independently. In this dissertation, the importance of location privacy of both the source and sink are addressed and four schemes called forward random walk (FRW), bidirectional tree (BT), dynamic bidirectional tree (DBT) and zigzag bidirectional tree (ZBT) respectively are proposed to protect them simultaneously. Simulation results illustrate the effectiveness of the proposed end-to-end location privacy protection schemes.

Keywords: Buffer Management, Cooperation Incentive, Delay Tolerant Networks, Resource Constraints, Routing, Security and Privacy.

PUBLICATIONS

Journal Papers

1. **Honglong Chen**, Wei Lou, Xice Sun and Zhi Wang. A Secure Localization Approach Against Wormhole Attacks Using Distance Consistency. *EURASIP Journal on Wireless Communications and Networking, vol. 2010, Article ID 627039, 11 pages, 2010. doi:10.1155/2010/627039.* 2010.

2. **Honglong Chen**, Wei Lou and Zhi Wang. A Novel Secure Localization Approach in Wireless Sensor Networks. *EURASIP Journal on Wireless Communications and Networking, vol. 2010, Article ID 981280, 12 pages, 2010. doi:10.1155/2010/981280.* 2010.

3. **Honglong Chen**, Wei Lou and Zhi Wang. On Providing Wormhole Attack Resistant Localization Using Conflicting Sets. *Submitted to Wireless Communications and Mobile Computing.* 2012.

4. **Honglong Chen**, Wei Lou, Junfeng Wu, Zhibo Wang, Zhi Wang and Aihua Xia. Label-Based DV-Hop Secure Localization Against Wormhole Attacks in Wireless Sensor Networks. *Submitted to Computer Communications.* 2012.

5. Zhibo Wang, Zhi Wang, **Honglong Chen**, Jianfeng Li, Hongbin Li and Jie Shen. Hier-Track: An Energy Efficient Cluster-based Target Tracking System for Wireless Sensor Networks. *Submitted to International Journal of Distributed Sensor Networks.* 2012.

6. Zhibo Wang, Wei Lou, Zhi Wang, Junchao Ma and **Honglong Chen**. HCTT: A Hybrid Cluster-based Target Tracking Protocol for Wireless Sensor Networks. *Submitted to Ad Hoc Networks.* 2012.

Conference Papers

1. **Honglong Chen** and Wei Lou. Group Aware Cooperative Routing for Opportunistic Networks under Resource Constraints. *Accepted to appear in the IEEE Global Communications Conference (**GLOBECOM 2012**)*. 2012.

2. **Honglong** and Wei Lou. On Using Contact Expectation for Routing in Delay Tolerant Networks. In *Proceedings of the* 40*th* *IEEE International Conference on Parallel Processing (**ICPP 2011**)*, pp. 683-692, 2011.

3. **Honglong Chen** and Wei Lou. From Nowhere to Somewhere: Protecting End-to-End Location Privacy in Wireless Sensor Networks. In *Proceedings of the* 29*th* *IEEE International Performance Computing and Communications Conference (**IPCCC 2010**)*, PP. 1-8, 2010.

4. **Honglong Chen**, Wei Lou and Zhi Wang. Secure Localization Against Wormhole Attacks Using Conflicting Sets. In *Proceedings of the* 29*th* *IEEE International Performance Computing and Communications Conference (**IPCCC 2010**)*, PP. 25-33, 2010.

5. **Honglong Chen**, Wei Lou and Zhi Wang. A Consistency-Based Secure Localization Against Wormhole Attacks in Wireless Sensor Networks. In *Proceedings of the International Conference on Wireless Algorithms, Systems and Applications (**WASA 2009**)*, PP. 368-377, 2009.

6. **Honglong Chen**, Wei Lou and Zhi Wang. Conflicting-Set-Based Wormhole Attack Resistant Localization in Wireless Sensor Networks. In *Proceedings of the International Conference on Ubiquitous Intelligence and Computing (**UIC 2009**)*, PP. 296-309, 2009.

7. **Honglong Chen** and Wei Lou. Making Nodes Cooperative: A Secure Incentive Mechanism for Message Forwarding in DTNs. *Submitted to IEEE International Conference on Communications(**ICC 2013**).* 2013.

8. Zhibo Wang, Zhi Wang, **Honglong Chen**, Jianfeng Li and Hongbin Li. Demo Abstract: An Energy-efficient Target Tracking System for Wireless Sensor Networks. In *Proceedings of the* 9*th ACM Conference on Embedded Networked Sensor Systems (**SenSys 2011**)*, pp. 377-378, 2011.

9. Zhibo Wang, Wei Lou, Zhi Wang, Junchao Ma and **Honglong Chen**. A Novel Mobility Management Scheme for Target Tracking in Cluster-based Sensor Networks. In *Proceedings of the IEEE International Conference on Distributed Computing in Sensor Systems (**DCOSS 2010**)*, pp. 172-186, 2010.

10. Junfeng Wu, **Honglong Chen**, Wei Lou, Zhibo Wang and Zhi Wang. Label-Based DV-Hop Localization Against Wormhole Attacks in Wireless Sensor Networks. In *Proceedings of the IEEE International Conference on Networking, Architecture, and Storage (**NAS 2010**)*, pp. 79-88, 2010.

11. Xice Sun, Zhi Wang, **Honglong Chen** and Wei Lou. A Probabilistic Model for lifetime measurement in privacy-aware sensor networks. In *Proceedings of the IEEE International Conference on Wireless Communications and Signal Processing (**WCSP 2009**)*, pp. 1-5, 2009.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere appreciation to my chief supervisor Dr. LOU Wei for his continuous encouragement and rigorous supervision during my whole Ph.D. research. Without his vision, advices, extensive knowledge, strong analytical skills and commitment to the excellence, this thesis would not have been completed. He unremittingly trained me to be a good researcher. His vision, passion, and attitude towards the research deeply affected my. What I have leaned and experienced will benefit me much in the future, including the study and life.

Also, I would like to thank and express my gratitude to the coexaminers of my guided study and confirmation defense, Dr. XIAO Bin and Dr. WANG Zhijun. They provided me with a lot of insightful comments and constructive suggestions to my research. I am also very grateful to my master supervisor Dr. WANG Zhi for his ceaseless encouragement and support during my Ph.D. research.

I would like to thank Dr. XIAO Qingjun and Mr. BAI Bing. They helped me to learn the TOSSIM and ONE simulations by sharing their valuable experience. I am also very grateful to all my coauthors, LI Hongbin, WANG Zhibo, MA Junchao, SUN Xice, WU Junfeng and LI Jianfeng, for their constructive suggestions. My thanks also go to all the members of our research group, YANG Libin, ZHANG Jin, XIONG Tao, JIAO Xianlong, LIU Nianbo and YAO Junmei, for their interesting discussions and suggestions. I would also like to thank the colleagues in my office, Kwong Ka Ming and Charles, for their assistance during my life in Hong Kong.

Finally, I would like to dedicate this thesis to my beloved wife, TIAN Tian, and my son, CHEN Zhuo. I hope I can make it up to them for all the time they sacrificed to help me to be devoted to my study. I would also like to express my gratitude to my parents, who give me unwavering faith and confidence in my study and life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Background

In most of conventional routing protocols [71, 29, 70], the messages are designed to be delivered along the always-connected end-to-end path between the source and destination with a very high delivery ratio. However, this kind of routing protocols is based on the assumption that the network topology is static and the end-to-end path always exists, which is not applicable in some challenged networks, such as the delay tolerant networks (DTNs) [18, 19, 27, 30, 49].

DTNs are an emergent communication paradigm, which consists of opportunistically connected mobile nodes. In DTNs, due to the sporadic node density, unpredictable node mobility and short radio range of the nodes, only the intermittent connectivity among the nodes exists and the network topology changes over time. In conventional networks, the nodes' mobility will be considered as an inconvenience that may cause the network partition, and the message delivery will be a very challenging issue due to the dynamic topology. However, in DTNs, the nodes can take advantage of the movement by sending the message over an existing link and buffering it at next hop until the next link in the path is available, i.e., the node carrying the message moves to the communication range of another node. This kind of message delivery process is referred as "store-carry-and-forward" [87] mechanism, which is always adopted in DTN routing.

DTNs are initially proposed for the environments such as the inter-planetary communications, in which the communication delay is relatively large due to the very long inter-planetary distance. Recently, DTNs have attracted the attentions of more and more researchers and can be competent for many applications such as social networks [45, 26], vehicular networks [5, 42, 40], pocket switched networks [73, 7] and habitat monitoring sensor networks [55, 90], etc. The widespread applications of DTNs will introduce more academic problems such as the routing, cooperation incentive, security and privacy, which need to be solved by researchers.

1.2 Motivation

As the nodes in DTNs are intermittently connected, it is very difficult, if not impossible, to determine a persistent source-to-destination path for each message. Thus, conventional routing protocols, which route messages along persistent end-to-end paths predesigned based on the always-connected network topology, are not applicable in DTNs. Although the store-carry-and-forward [87] mechanism is always adopted in DTNs to take advantage of the nodes' mobility to deliver messages, it is still hard for a node to obtain the global network connectivity as it is time-varying, which makes the routing very difficult. Thus, this dissertation has the motivation to design an efficient routing protocol to deliver messages in DTNs.

Due to the high mobility of the nodes in DTNs, the contact duration between each pair of encountering nodes is limited, which may be not long enough for them to exchange messages. It is necessary for two encountering nodes to cooperate with each other to schedule the message transfer sequence, which can efficiently make use of the limited bandwidth. Also, the nodes in DTNs are always some portable devices such as smart phones or PDAs, which are usually resource-limited (e.g., the limited buffer space). To obtain a high delivery ratio in DTNs, each message may have multiple copies in the network, which are buffered by multiple nodes. Consequently, a buffer management strategy is required to schedule the message dropping when the buffer is full in this kind of networks. Thus, the second motivation of this dissertation is to design a cooperative routing protocol for the DTNs under resource constraints of the bandwidth and buffer space.

In the traditional DTN routing protocols, it is always assumed that each individual node is willing to forward messages for others. However, this assumption may easily be violated in the non-cooperative DTNs, in which the traditional DTN routing protocols will be impractical. In the non-cooperative DTNs, the nodes may be managed by some rational individuals, such as human-beings or other autonomous parties. Such nodes may be selfish [6, 8], i.e., they may decline to cooperate with others to truthfully forward the messages if they cannot make any profit from the message forwarding. Some incentive mechanisms [25, 94, 41], which requires the path from the source to destination to be stable for most of the time, have been proposed to motivate the cooperation for the nodes in the wireless ad hoc networks. However, this kind of mechanisms cannot be applicable in the DTNs since the nodes are intermittently connected and the path from the source to destination is not always existing. Due to the distributed characteristic of DTNs, it is a challenging issue to detect and prevent the selfish behaviors conducted by the individual nodes in the network. Therefore, to make the DTN routing protocols applicable in the non-cooperation DTNs, this dissertation has the motivation to to design the efficient incentive mechanisms to motivate the selfish nodes to cooperate with each other in the message forwarding.

In the delay tolerant event collection systems [90, 24], which are one of the typical DTN applications, the sensor node who detects the occurrence of an interesting event will send the information to the sink or base station via multi-hop wireless communications. As the wireless channels can be accessed by anyone who wishes, it is not difficult to attack the wireless networks. Thus, if the delay tolerant event collection systems are applied to monitor some rare animal such as the panda, it will be unsafe since an attacker is easily able to either locate the source by back tracing hop-by-hop to capture the monitored animal or locate a receiver by following the flow of packets in the network to destroy the sink or based station, which will make the whole event collection system crash. Thus, the end-to-end location privacy protection is a very important problem in the delay tolerant event collection systems. Previous location privacy protection schemes can only protect the source-location privacy or the sink-location privacy independently, which motivates this dissertation to propose the location privacy protection schemes to protect the location privacy of both the source and sink in the delay tolerant event collection systems.

To sum up, this dissertation has motivation to:

- Design an efficient routing protocol to deliver messages in DTNs.
- Design a cooperative routing protocol in the DTNs under resource constraints of both the bandwidth and buffer space.
- Design a cooperation incentive mechanism to motivate the nodes to cooperate with each other to forward the messages in the non-cooperative DTNs.
- Design a location privacy protection scheme to protect the end-to-end location privacy in the delay tolerant event collection systems, which are one of the typical DTN applications.

1.3 Contributions of the Thesis

As shown in Figure 1.1, the contributions of this thesis mainly focus on three topics: 1) propose the traditional DTN routing protocols; 2) propose the cooperation incentive mechanisms to motivate nodes in message forwarding in the non-cooperative DTNs; and 3) propose the location privacy protection schemes to protect the location privacy of both the source and sink in the delay tolerant event collection system which is one of the applications in DTNs. This section will introduce the contributions of this thesis from the three topics respectively.

1.3.1 Routing Protocols in DTNs

The routing protocols in DTNs can be classified into three categories: epidemic-based [84], forwarding-based [27] and quota-based [57]. As the quota-based routing protocols can well balance the tradeoff between the epidemic-based routing and the forwarding-based routing on the performance of delivery ratio and overhead, it will be adopted as the basis of the proposed DTN routing protocols in this dissertation.

1.3.1.1 Routing Protocols Using Contact Expectation

One potential solution for the DTN routing is to use the contact expectation to estimate the future contact information. Embedded this idea, this dissertation proposes an expected encounter based routing protocol (EER) to solve this problem. EER has two phases: multiple replicas distribution and single replica forwarding. In the multiple replicas distribution phase, each node disseminates the replicas of a message to different nodes as soon as possible, which can be achieved by distributing the replicas of the message according to their expected encounter values (EVs). The expected EV is calculated as a function of the message's time-to-live (TTL), which is more accurate in predicting the future EV in a fixed future time interval. In the single replica forwarding phase, each node decides whether to forward the message to its current encounter by comparing their minimum expected meeting delays (MEMDs) to the destination. The MEMD is calculated based on the past meeting intervals between each pair of nodes and the elapsed time since their last contact.

A community based routing protocol (CR), which takes advantages of the high contact frequency property of the community, is further proposed. CR includes inter-community routing and intra-community routing. In the inter-community routing, each node disseminates the multiple replicas of a message to the nodes from different communities as soon as possible, in which the distribution of the replicas of this message is proportional to the expected numbers of encountering communities (ENECs) of any pair of encounters. In case of the single replica of the message left during the propagation, the message is delivered to the node which has a higher probability to encounter the destination community. In the intra-community routing, a node in the destination community distributes the replicas of a message to its encounter in the same community according to the proportion of their expected EVs within the community. In case of the single replica of the message, the node in the destination community decides whether to forward the message to its encounter in the same community by comparing their MEMDs within the community, which leads the message to be forwarded to its final destination. The EER and CR protocols do not consider the buffer management, and the FIFO (first-in-first-out) scheme is adopted.

The proposed EER and CR protocols are evaluated in the ONE simulator to demonstrate their effectiveness. The effects of the network parameters on the performance of the proposed routing protocols are also analyzed.

1.3.1.2 Group Aware Routing Protocol under Resource Constraints

DTNs can be formed by mobile nodes such as the portable devices carried by human beings [69]. In such DTNs, mobile nodes with common interest or close relationship tend to form into groups and move together. One typical DTN routing application scenario is the disaster recovery system [58]. For example, an intense earthquake may devastate most infrastructures of cellular networks, making the communication services broken down for a long time. In many post-earthquake scenarios, survivors would tend to move in groups to assist each other in case of further disasters. Thus, the mobile phones carried by moving people can construct a DTN with group feature to provide emergent communication services. Under such emergent situations, the survivals would likely try to contact their relatives or friends as early as possible, which will bring about a huge communication workload for the network and lead to the stringent constraints of the bandwidth and buffer space for the mobile nodes. This challenging scenario motivates us to design a group aware routing protocol for the DTNs under resource constraints. As the nodes within a group likely encounter each other more frequently and the connectivity among the group members is relatively stable, the source node can apply the strategy by first delivering the message to the destination group, and then letting the message be routed to the destination using the intra-group links. Moreover, the nodes can also cooperate with their group members to share their limited buffers in caching the messages more efficiently than the FIFO scheme in the EER and CR protocols.

In this dissertation, a cooperative routing protocol using the group feature is proposed, the idea behind which is to maximize the message delivery probability in the DTNs under resource constraints, i.e., the limited bandwidth and buffer space. The proposed routing protocol includes a cooperative message transfer scheme and a buffer management strategy. In the cooperative message transfer scheme, the limited bandwidth available for mobile nodes is considered and two encountering nodes will exchange messages cooperatively to maximize the delivery probability. In the buffer management strategy, the constraint of mobile nodes' buffer space is further considered, and the cooperative message caching scheme, in which the message dropping priorities are designed to minimize the reduced delivery probability, is proposed. The proposed cooperative routing protocol is implemented in the ONE simulator and the simulations are conducted to illustrate its effectiveness under different network parameters.

1.3.2 Cooperation Incentive Mechanisms for Message Forwarding in Non-cooperative DTNs

Generally, the incentive mechanisms can be classified into two categories [83]: reputation-based schemes and credit-based schemes. The reputation-based schemes require each node to monitor the traffic information of all its neighbors and keep track of their reputation values, which should be propagated to all other nodes efficiently and effectively. This is quite a challenge for DTNs due to the intermittent connectivity among the nodes. On the other hand, the credit-based schemes use virtual credits to motivate selfish nodes to participate in the message forwarding; the credits they earned from forwarding other nodes' messages can be used to pay for the delivery of their own ones. It is obvious that the credit-based schemes better adapt to the intermittent connective characteristic of DTNs and will be adopted in this dissertation.

The design goals for the cooperation incentive mechanisms in this dissertation are three-fold: 1) incentive compatibility: to make the truthful forwarding be the dominant strategy for all the nodes; 2) budget control: to guarantee that the payment for each delivered message is upper bounded; 3) security enhancement: to defend against the typical attacks in the cooperation incentive mechanisms. By considering the design goals, a credit-based rewarding scheme called earliest path singular rewarding (EPSR) scheme is first proposed to motivate the nodes to truthfully forward the messages during every contact opportunity. By further considering that a node may get more contact information of others and misbehave accordingly to take advantage of that, another credit-based rewarding scheme called earliest path cumulative rewarding (EPCR) scheme is then proposed. The main idea of the proposed rewarding schemes is to reward each node in the earliest delivery path according to its *contribution time*, which is the period of time that the node holds the message. The simulations based on the real trace are conducted to analyze the effects of the selfish nodes on the routing performance and illustrate the effectiveness of the proposed rewarding schemes.

1.3.3 Location Privacy Protection Schemes in Delay Tolerant Event Collection Systems

The end-to-end location privacy protection is a very important problem in the delay tolerant event collection systems, which are one of the typical DTN applications. Lots of location privacy protection schemes for the routing in the delay tolerant event collection systems have been developed in the past decade. However, these proposed schemes can only protect the source location privacy or the sink location privacy independently.

In this dissertation, four end-to-end location privacy protection schemes are proposed to protect against a local eavesdropper who might breach the location privacy of a source or sink, that is, end-to-end location privacy. The four schemes are forward random walk (FRW), bidirectional tree (BT), dynamic bidirectional tree (DBT) and zigzag bidirectional tree (ZBT). In the forward random walk scheme, every node relays a received message to a node randomly chosen from its forward neighbors whose hop-count to the sink is no larger than its own. This procedure is repeated at each node until the message arrives at the sink. To increase the location anonymity, tree topology is employed at the two ends of the routing path respectively in the bidirectional tree scheme. In the dynamic bidirectional tree scheme, branches of the trees are generated dynamically, which can improve the performance. However, in the bidirectional tree scheme, real messages are routed along the shortest path, which makes it possible for an eavesdropper to infer the location of the source and sink by extending the line of the shortest path. To solve this potential threat, a proxy source and a proxy sink are devised in the zigzag bidirectional tree scheme, making it more difficult for the adversary to obtain the location of the source or sink. The effectiveness of the proposed location privacy protection schemes are illustrated with TOSSIM-based simulations.

1.4 Outline of the Thesis

The structure of this thesis is described in Figure 1.1. Chapter 1 is the introduction to this thesis. Chapter 2 briefly presents the literature review on DTN routing protocols, cooperation incentive mechanisms and location privacy protection schemes respectively. The traditional DTNs are considered in Chapter 3 and Chapter 4. In Chapter 3, the routing protocols using contact expectation in DTNs including the expected encounter based routing protocol and the community based routing protocol are proposed. In Chapter 4, the group aware cooperative routing for the DTNs under resource constraints of the bandwidth and buffer space is proposed. In Chapter 5, the non-cooperative DNTs are considered and two secure credit-based rewarding schemes are proposed to motivate the nodes to truthfully forward messages. Chapter 6 proposes four end-to-end location privacy protection schemes in the delay tolerant event collection sys-

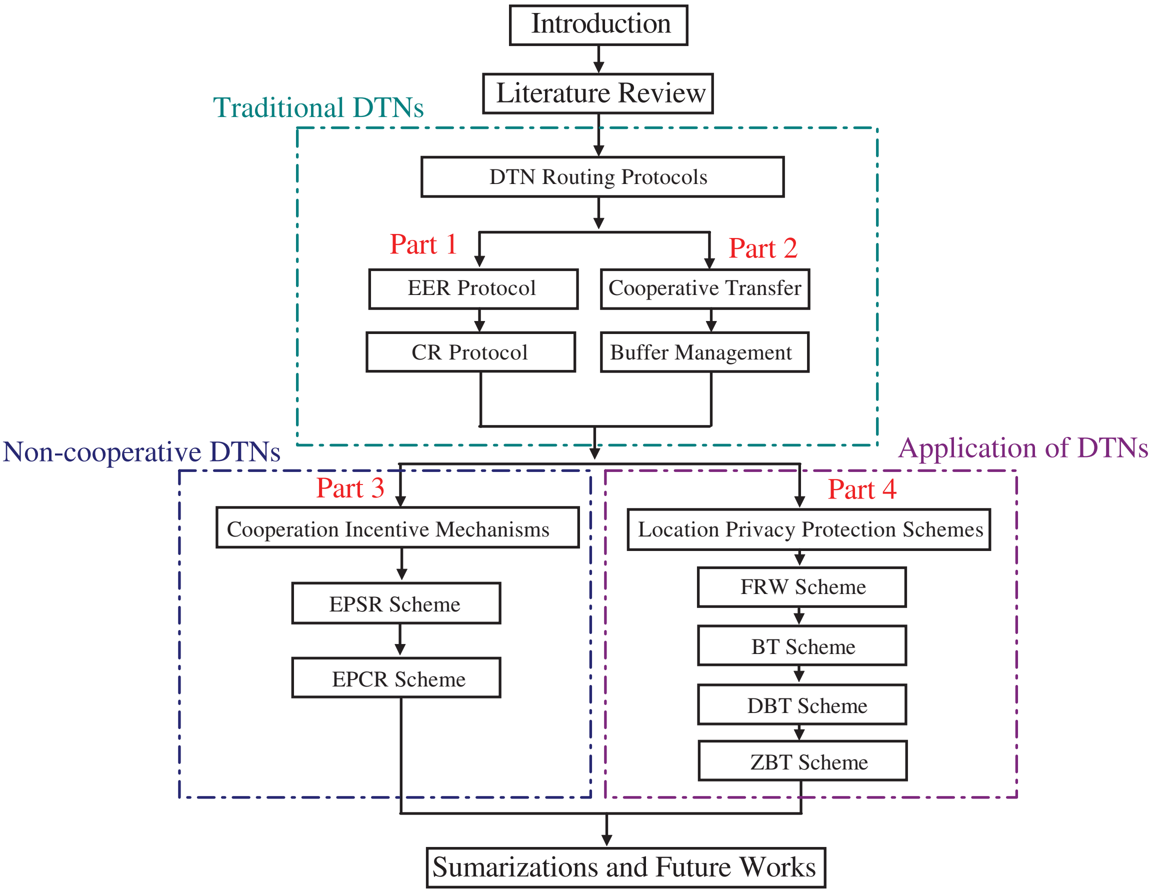


Figure 1.1. An outline of this thesis.

tems, which are one of the typical DTN applications. Chapter 7 summarizes this thesis and puts forward the future works.

CHAPTER 2
LITERATURE REVIEW

DTNs have attracted the attentions of the researchers during the past few years and a lot of related work has been proposed. This chapter will introduce the previous related work on three topics respectively, i.e., DTN routing protocols, cooperation incentive mechanisms and location privacy protection schemes.

2.1 DTN Routing Protocols

Routing protocols in DTNs attempt to deliver the message through the intermittently connected nodes to the destination. The property of contemporaneous links makes routing in DTNs a challenging issue. The DTN routing has become an active research topic and many protocols [2, 4, 8, 46, 74, 80, 93, 48] have been proposed in the past few years. This section will discuss the typical DTN routing, social networks based DTN routing, buffer management strategies and community detection schemes respectively.

2.1.1 Typical DTN Routing

The typical routing protocols in DTNs can be classified into three categories: *epidemic-based*, *forwarding-based* and *quota-based* routing protocols.

2.1.1.1 Epidemic Based Routing

To obtain a high message delivery ratio, Vahdat *et al.* propose the epidemic routing protocol [84], in which each message is replicated and flooded to all the nodes in the network. However, in this protocol, the number of replicas of each message in the network increases rapidly, which greatly consumes the limited buffer space and bandwidth. To reduce the overhead, some improved epidemic-based routing schemes [46][5][16] are proposed. In Prophet [46], each node maintains the delivery probabilities to the other nodes and it will replicate and forward a message to its encounter only if its encounter has a higher delivery probability to the destination. In MaxProp [5], the most likely to be successfully delivered packets will be replicated and transmitted to other nodes with the highest priority. In the delegation forwarding [16], the message is replicated and delivered by a node to its encounter only if its encounter has a better quality metric, such as success rate, average delay, and cost. This method can reduce the cost of the network to $O(\sqrt{n})$, compared to the cost $O(n)$ of the epidemic routing, where n is the number of nodes in the network.

2.1.1.2 Forwarding Based Routing

To avoid a high network overhead, some forwarding-based routing schemes with single copy [27, 30, 92] are also proposed. In [27], the delay tolerant network routing problem is formulated as a connectivity graph that is time-varying and they assume some of the network dynamics is known in advance, based on which the delivery path of the message can be designed. Several routing algorithms corresponding to the percentage of knowledge of the network dynamics are then proposed. Jones *et al.* [30] design a practical single copy routing mechanism based on the minimum estimated expected delay, which is calculated based on the average meeting interval between each pair of nodes in the network using the Dijkstra's algorithm. While it, as a link-state routing algorithm, requires the pair-wise routing information exchange among encountering nodes, which introduces additional overhead, it can provide the complete topology at each node to achieve good performance. The predict and relay (PER) [92] scheme relies on two observations: 1) nodes usually move around a set of well-visited landmarks points; 2) node mobility behavior is semi-deterministic and could be predicted once there is sufficient mobility history information. The PER scheme models the mobility of a node as a time homogeneous semi-markov process, based on which it can predict the future contacts between each pair of nodes for the message delivery.

2.1.1.3 Quota Based Routing

A tradeoff between the epidemic-based routing and the forwarding-based routing with single copy is the quota-based routing protocols [78, 79, 50, 42, 57], which implant a predefined number of replicas of each message into the network to improve the delivery ratio without greatly increasing the overhead. Spray-and-Wait [78] disseminates the replicas of each message in the *spray* phase. When the node has only one replica of the message, it will transform into the *wait* phase, in which it waits to meet the destination and then delivers the message directly to the destination. Spray-and-Focus [79] adopts the *spray* phase in [78], and when the node has only one replica of the message, it will transform into the *focus* phase, in which the message can be forwarded to its encounter with a higher utility to improve the performance. Based on Spray-and-Wait, Liu and Wu propose an optimal probabilistic forwarding protocol [50] to maximize the expected delivery rate, in which each forwarding is modeled as an optimal stopping rule problem. However, it assumes each node knows the mean inter-meeting times of all pairs of nodes in the network, which is impractical. In [42], the DTN routing approach is proposed, in which nodes form predicted opinions towards future contacts by utilizing the time information recorded in the encounter tickets. The proposed routing approach is blackhole attacks resistent by using the encounter tickets to secure the evidence of each contact. CREST [81] proves that the contact behavior between individual pairs follows the log-normal distribution and the the conditional residual time, which indicates the time remaining before node i and j meet conditioned on the information that they last met t_{ij} time slots ago, is used in the proposed forwarding protocol to obtain a high delivery ratio. In [22], the traces are divided into "*on-period*" and "*off-period*". The contact process during the on-period is much more stable and predictable, thus it will be used on the contact prediction for routing. An encounter based routing scheme called EBR is proposed in [57], which distributes the replicas of a message between two encounters according to the proportion of their estimated EVs. However, the estimated EV in [57] is identical to all messages and independent of the TTL of each message.

2.1.2 Social Networks Based DTN Routing

The small world dynamics have been proposed for the economics and social studies, and the researchers have proved that some properties of the social networks can be well utilized in the DTN routing. One of the most important properties in the social networks, which can be employed in the DTN routing, is the community. Thus, the community detection is the prerequisite for the community based DTN routing. There has been considerable recent interest in algorithms for finding communities in networks. In [61], the community detection approaches are divided into two categories: the computer science approaches and the sociological approaches. The computer science approaches generally consider the community detection as a *graph partitioning* problem and try to find some graphic theoretical solution to solve this problem. The principal technique in the sociological approaches is to use *hierarchical clustering*. Newman *et al.* in [62] propose a set of algorithms for discovering community structure in networks, which share two definitive features. The first feature is to iterative remove edges from the network to split it into communities and the edges removed are identified using a "betweenness" measure. The second feature is to recalculate the betweenness measure after each removal. In [59], it is pointed out that the weighted networks can be analyzed using a simple mapping from a weighted network to an unweighted multigraph, thus the standard techniques for detecting community structure for unweighted graphs can be applied to the weighted ones. In [66], it is found that most of the actual networks are made of highly overlapping cohesive groups of nodes. However, the previous community deterministic methods are only suitable to find separated communities. Thus, Palla *et al.* in [66] introduce an approach to analyze the main statistical features of the interwoven sets of overlapping communities, after which they apply an efficient technique for exploring overlapping communities on a large scale network. In [89], a community detection algorithm that takes into account the aging and weight of contacts between mobile entities is presented to identify communities that are dynamically changing in mobile environment.

In social networks, the centrality [20, 21] of a node is a measure of its social importance. The similarity and betweenness, which are calculated based on the centrality, are employed in the DTNs as the utility metrics in SimBet Routing [12]. The proposed SimBet Routing exploits the exchange of pre-estimated betweenness centrality metrics and locally determined social similarity to the destination node. BUBBLE [26] is a social-based forwarding algorithm using the properties of social network and it is designed for pocket switched networks. In BUBBLE, a node first bubbles the message up the hierarchical ranking tree using the global ranking, until it reaches another node, which is in the same community as the destination node. Then the local ranking system is used instead of the global ranking, and the message continues to bubble up through the local ranking tree until the destination is reached or the message expires. In [23], the multicast in DTNs is well studied from the social network perspective, in which the relay selections for multicast are formulated as a unified knapsack problem by exploiting node centrality and social community structures.

2.1.3 Buffer Management Strategies

The carry-store-and-forward mechanism is always adopted in the DTN routing. And the message would be replicated, which results in multiple replicas of a message existing in the network. Thus, for the nodes with limit buffer space, the buffer management is necessary and the researchers have already proposed some schemes. In MaxProp [5], due to the limitation of buffer space, each node schedules the packets to be dropped and it will drop the packet with the smallest probability to be successfully delivered when the buffer is full. In [36] and [35], efficient buffer management policies for delay tolerant networks, which can be tuned to minimize the average delivery delay or to maximize the average delivery ratio, are proposed. However, the proposed policies only consider the direct forwarding from the current node to the destination, which may be not suitable in DTNs with intermittent connectivity. Lindgren et al. [47] evaluate a set of combinations of existing buffer management strategies and routing protocols for DTNs, in which they show that Prophet routing [46] can achieve better performance when adopting the right buffer management strategy. Li *et al.* [42] propose the adaptive optimal buffer management policies for realistic DTN, in which the mobility model is adjusted according to the nodes' historical meeting information, and it can optimize certain network performance goals, such as maximizing the average delivery ratio or minimizing the average delivery delay. In [39], a routing scheme for socially selfish DTNs is proposed, which takes the size of each message into consideration. However, this scheme is a bit unfair since it attempts to deliver the message with smaller size and to drop the message with larger size while the delivery ratio is calculated in the term of the number of delivered messages.

2.2 Cooperation Incentive Mechanisms

To motivate the cooperation among the selfish nodes in the wireless networks, many incentive schemes [25, 83, 94, 8, 10] have been proposed. The incentive schemes for the message forwarding in the wireless networks can be generally classified into the reputation-based schemes and the credit-based schemes [83].

2.2.1 Reputation Based Incentive Schemes

In the reputation-based incentive schemes, each node will monitor its encounters' traffic information and keep track of the reputation values of all the other nodes. The reputation of a node will increase when it forwards a message, which is eventually delivered successfully. Based on the node's reputation, differential services will be provided. In [25], a secure and objective reputation-based incentive scheme called SORI is proposed to encourage the node to forward the message truthfully and discipline the node's selfish behaviors, in which the reputation of a node is quantified by objective measures and the propagation of reputation is efficiently secured by a one-way-hash-chain-based authentication scheme. CONFIDANT [3] is proposed to make the misbehavior unattractive, which is based on selective altruism and utilitarianism. In CONFIDANT, nodes learn the malicious behavior not only from their own experience, but also from observing the neighborhood and from the experience of their friends. It aims at detecting and isolating the misbehaving nodes to make it unattractive for each node to decline the cooperation. Uddin et al. propose a rewarding scheme called RELICS [83], which relies on a simple principle — *a node's message is forwarded only if it has forwarded messages originated from others.* The proposed RELICS provides incentive to nodes in a physically realizable way in that the rewards are reflected into network operations. The ranking (reputation) of nodes depending on their transit behaviors is employed and those ranks are translated into the message priority in the forwarding.

2.2.2 Credit Based Incentive Schemes

The credit-based incentive schemes motivate each node to help in forwarding messages of others to earn virtual credits, which it can use when it wants to deliver its own generated messages. A Trusted Third Party (TTP) is a necessity in the credit-based incentive schemes to manage the rewarding procedure. Zhong et al. [94] propose an incentive scheme called Corsac based on the VCG mechanism. Corsac applies efficient cryptographic techniques to design a forwarding protocol to enforce the routing decision, such that fulfilling the routing decision is the optimal action of each node in the sense that it brings the maximum utility to the nod. Another VCG-based incentive scheme proposed in [41] uses a principal-agent model to motivate the message forwarding in multi-hop wireless ad hoc networks with hidden information and actions, in which several algorithmic mechanisms are designed for a number of routing scenarios such that each selfish agent will maximize its utility (i.e., profit) when it truthfully declares its type (i.e., cost and its actions) and it truthfully follows its declared actions. However, the VCG-based incentive schemes are known to suffer from the sybil attack and they cannot guarantee that the payment for each delivered message is finite. In SMART [95], the layer coin, including the *based layer* and *endorsed layer*, is employed in the proposed incentive scheme, which can be implemented in a fully distributed manner to thwart various attacks without relying on any tamperproof hardware. However, the proposed SMART can defend against several attacks but not the sybil attack. Li and Wu propose an incentive scheme for vehicular networks called FRAME [38] that uses a forwarding tree to represent the message propagation process and allocate weighted rewards to the intermediate nodes according to their positions in the forwarding tree. FRAME also introduces a sweepstake method in the incentive scheme, which can attract more nodes to participate in forwarding. Furthermore, the security measurements to secure the evidence required by the incentive scheme are developed. In PIS [54], a practical incentive scheme based on micropayment is proposed for traditional multihop wireless networks to stimulate the nodes' cooperation and a reactive receipt submission mechanism is proposed to reduce the number of submitted receipts and protect against collusion attacks. In INPAC [9], an enforceable incentive scheme using a combination of game theoretical and cryptography techniques is proposed, in which it is proved that if INPAC is used, then following the protocol faithfully is a subgame perfect equilibrium. The MobiCent proposed in [8] is a credit-based incentive system that motivates the epidemic routing in DTNs, which can defend against the edge insertion attack and edge hiding attack. The proposed MobiCent also provides different payment mechanisms to cater to client that wants to minimize either payment or data delivery delay. While it cannot solve the scenario when two nodes collude together to reduce the hop count of a path to cheat the system for the extra reward.

2.3 Location Privacy Protection Schemes

Anonymity, which is defined by Pfitzmann [72] as “the state of being not identifiable within a set of subjects”, can be used to protect the location privacy in the traditional networks. Reed [52] designs an application-independent anonymous connecting solution called Onion Routing, which consists of many onion routers. Data is transmitted through a path of onion routers and is encrypted multiple times using the symmetric keys distributed to all the onion routers on the path. Each onion router removes or adds a layer of encryption depending on the direction of the data for the anonymous routing. The internet-based anonymous system Tor [75] randomly chooses a serial of onion routers from a set of well-organized multiple directory servers, which provides the traffic flow confidentiality by enclosing a great deal of second-generation onion routers to relay the data traffic.

In the habitat monitoring sensor networks, there are many sensor nodes deployed in the network to detect the occurrence of the specified events, after which they can send the information to the sink. In this kind of event collection systems [90, 24], the location privacy of the source and sink is also very important. Previously proposed location privacy protection schemes only focus on the source location privacy or the sink location privacy. In this section, the source location privacy and the sink location privacy will be discussed respectively.

2.3.1 Source Location Privacy Protection Schemes

Many schemes have been proposed for the protection of the source location privacy in the event collection systems. [31] and [65] have proposed a source location privacy scheme that makes use of the *Panda-Hunter* problem as an application scenario for monitoring-oriented sensor networks where the location privacy is important. It has been proven that the *Phantom routing* protocol can make use of a random walk to prevent attackers from identifying the source, while not incurring a noticeable increase in energy overhead. Xi *et al.* [88] have proposed a two-way random walk routing protocol (from both source and sink) called GROW (greedy random walk), which can reduce the opportunity for an eavesdropper to collect the location information. In GROW, the delivery rate can be improved by using local broadcasting and greedy forwarding. Although the message delivery time is a little longer than that of the broadcasting-based approach, it is still acceptable when considering its enhanced privacy preserving capability. Also, the energy consumption of GROW is less than half the energy consumption of flooding-base phantom routing, which is preferred in a low duty cycle, environmental monitoring sensor network. PRLA [86] protects the source location privacy by using so-called inclination angles to ensure that every random walk gets away from the region close to the source. During the message delivery, PRLA chooses the next hops in random walk area with different probabilities, which can optimize the routing path and greatly increase the safety period. In [63], loops are generated in the network. When a message is routed along a path from the source to the base station and it encounters one of these pre-configured loops, the encountered loop will be activated and will begin cycling fake messages around the loop. The adversary has to go around these loops, thereby being led away from the real path, which guarantees a high safety period, i.e., the adversary requires a high expected time to locate a source node. A suboptimal privacy routing scheme called WRS has been proposed in [85] to protect the source location privacy by distributing message flows to different disjoint routes. It also formulates the performance bound for any routing scheme. Li *et al.* [43, 44] protect the source location privacy through a two-phase routing process. In the first phase, the packet travels randomly through the intermediate nodes before it is routed to a ring node. Then the packet is mixed with other packets through a network mixing ring (NMR). In [56], two techniques called *periodic collection* and *source simulation* are proposed. In the periodic collection, every node sends messages periodically, making the network n-anonymous. In the source simulation, it provides trade-offs between privacy, communication cost and latency. Four schemes named *naive*, *global*, *greedy* and *probabilistic* are proposed in [64] to provide location privacy against a laptop-class attack. In *naive* scheme, the maintenance messages are sent periodically to hide the real event reports. The *global* and *greedy* schemes improve the *naive* scheme by reducing the latency of event delivery but not increase the communication overhead. The *probabilistic* scheme further improves the performance by reducing the communication overhead without sacrificing the location privacy. Yang *et al.* [91] propose to use proxies to protect the source location of an event, the idea of which is to introduce carefully chosen dummy traffic to hide the real event sources in combination with mechanisms to drop dummy messages to prevent explosion of network traffic. A prototype of the scheme is implemented on Mica2 motes. In [76] FitProbRate is proposed, which first adopts the statistically strong source anonymity to reduce the latency efficiently.

2.3.2 Sink Location Privacy Protection Schemes

Sink location privacy has also been well studied. In [13], Deng *et al.* have proposed a base station privacy scheme against the traffic-rate analysis attack that randomly delays the transmission time of each message. They have also proposed in [14] to defend against the traffic analysis attacks. To achieve the location privacy of the base station, they first introduce a degree of randomness in the multi-hop path a packet takes from a sensor node to a base station, based on which the random fake paths are generated to confuse an adversary from tracking a packet as it moves towards a base station. Furthermore, multiple, random areas of high communication activity are created to deceive an adversary as to the true location of the base station. LPR [28] provides the receiver location privacy against the message tracing attacks. In LPR, the message flows incoming from and outgoing to a sensor node are uniformly distributed, which makes it difficult for an adversary to ascertain the direction of the sink. Moreover, LPR injects fake messages into the network to get a longer safety period. In [15], the sink anonymity is achieved by omitting the address of the sinks in routing, which can prevent the attackers from obtaining the receiver address by capturing the destination field of the packets. The proposed scheme can also prevent the attackers from predicting the location of the sinks by observing the flow of network traffic.

However, the common drawback of all these approaches is that they only consider the location privacy of the source or sink independently, while it is particular important to protect the location privacy of both the source and sink simultaneously for some DTN application scenarios such as the delay tolerant event collection systems. This dissertation will combine the protection of the source location privacy and sink location privacy in the delay tolerant event collection systems together.

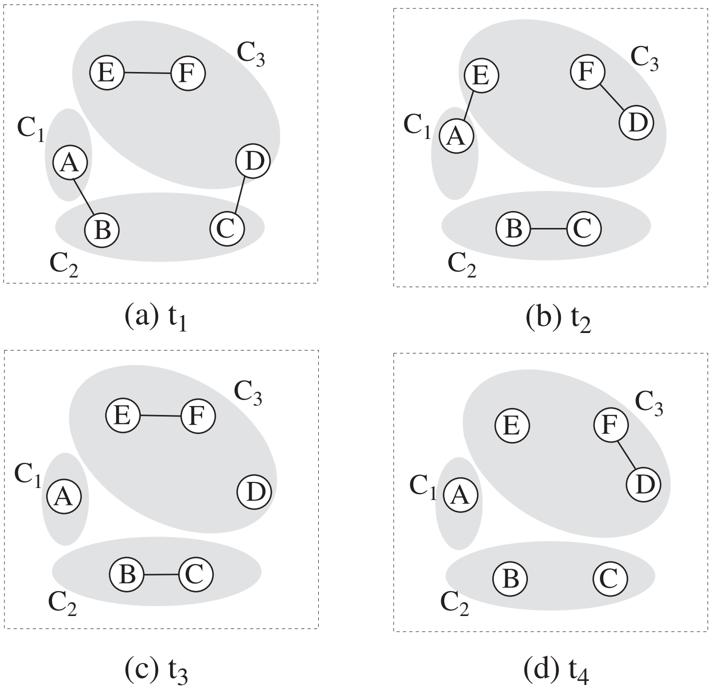
CHAPTER 3
ROUTING PROTOCOLS USING THE CONTACT EXPECTATION

3.1 Introduction

In a conventional network, messages are routed along persistent end-to-end paths predesigned on the always-connected network topology. However, this kind of routing strategy is not applicable to delay tolerant networks (DTNs), since the predesigned end-to-end path does not always exist. As the nodes may be mobile, the contact between each pair of nodes is intermittent and the network topology changes over time. Therefore, it becomes a most challenging task to design an efficient routing protocol in DTNs.

As the nodes in a DTN are intermittently connected, it is very difficult, if not impossible, to determine a persistent source-to-destination route for each message. The nodes can adopt the store-carry-and-forward mechanism to deliver the messages. However, it is still hard for a node to obtain the global network connectivity as it is time-varying. Figure 3.1 shows a simple network with six nodes. The network topology varies from time t_1 to t_4, making the routing in this network challenging. For instance, if node A wants to send a message to node D at t_1, according to the global network connectivity, the optimal path for this message is from node A to node E at t_2, then from node E to node F at t_3 and finally from node F to node D at t_4. However, node A may apply the best effort strategy to deliver the message to node B at t_1 since it meets node B firstly, resulting in failing to deliver the message to node D finally. Fortunately, by referring to the historical mobility, a node can predict its future contacts with other nodes, which are useful in making routing decisions.

A promising solution is proposed in [57] that predicts nodes' future contacts based on their



(a) t_1 (b) t_2 (c) t_3 (d) t_4

Figure 3.1. A sample delay tolerant network with six intermittently connected nodes. C_1, C_2 and C_3 denote three different communities in the network.

contact histories: Each node estimates its future encounter value (EV) based on its contact history. When two nodes meet, the replicas of a message are distributed between them according to the proportion of their estimated EVs. This approach can achieve good performance with a low overhead. However, the EV estimated in [57] is identical to all messages and independent of the time-to-live (TTL)[3.1] values of the messages. Since each message has its own TTL and should be delivered to the final destination before its TTL expires, the TTL of the message should be taken into consideration for estimating EV. For example, if a node estimates its EV as *e* per day, which suggests that this node will meet *e* other nodes in one day. However, if the residual TTL of the message is only one hour, then it is unwise to make the replicas distribution according to *e*. A better solution is to predict the EV based on the message's TTL.

Embedded this idea, in this chapter, an expected encounter based routing protocol (EER)

[3.1] The TTL used in this chapter is measured in the scale of time unit but not hop count.

is proposed to solve this problem. EER has two phases: multiple replicas distribution and single replica forwarding. In the multiple replicas distribution phase, each node disseminates the replicas of a message to different nodes as soon as possible, which can be achieved by distributing the replicas of the message according to their expected EVs. The expected EV is calculated as a function of the message's TTL, which is more accurate in predicting the future EV in a fixed future time interval. In the single replica forwarding phase, each node decides whether to forward the message to its current encounter by comparing their minimum expected meeting delays (MEMDs) to the destination. The MEMD is calculated based on the past meeting intervals between each pair of nodes and the elapsed time since their last contact. A community based routing protocol (CR), which takes advantages of the high contact frequency property of the community, is further proposed. CR includes inter-community routing and intra-community routing. In the inter-community routing, each node disseminates the multiple replicas of a message to the nodes from different communities as soon as possible, in which the distribution of the replicas of this message is proportional to the expected numbers of encountering communities of any pair of encounters. In case of the single replica of the message left during the propagation, the message is delivered to the node, which has a higher probability to encounter the destination community. In the intra-community routing, a node in the destination community distributes the replicas of a message to its encounter in the same community according to the proportion of their expected EVs within the community. In case of the single replica of the message, the node in the destination community decides whether to forward the message to its encounter in the same community by comparing their MEMDs within the community, which leads the message to be forwarded to its final destination.

3.2 Expected Encounter Based Routing Protocol

In this section, the proposed expected encounter based routing protocol (EER), which adopts the link-state routing [30] and is one of the quota-based routing protocols, is described. The

EER includes two phases: the multiple replicas distribution and the single replica forwarding. The multiple replicas distribution phase and the single replica forwarding phase will first be described in detail respectively, after which the EER algorithm will be elaborated.

3.2.1 Multiple Replicas Distribution Phase

In the EER, each message in the network is initiated with a predefined number of replicas. In the multiple replicas distribution phase, a node holds more than one replica of a message. To achieve a high message delivery ratio, the node can disseminate the replicas to different nodes as soon as possible. Therefore, when a node encounters any other node, it splits the replicas between them proportionally according to their expected EVs in a fixed future time interval, which can be calculated based on their contact histories.

3.2.1.1 Expected Encounter Value

As the previous work [30][77] has shown that the mobility observations can make predictions with a very large accuracy, each node can make a prediction based on its previous contact history. According to the node's contact history, it can predict its future contact information between itself and any other node. One of such contact information is the expected EV, i.e., the number of nodes a node expects to meet, which will be used in the replicas distribution.

To calculate the expected EV, each node needs to record the encounter time of each contact between itself and any other node. Assume that there are total n nodes in the network. Each node maintains a set of sliding windows to record the contact histories, e.g., the past meeting intervals between itself and any other encountering node. The set of recorded past meeting intervals between nodes u_i and u_j is $R_{ij} = \{\Delta t_1^{ij}, \Delta t_2^{ij}, ..., \Delta t_{r_{ij}}^{ij}\}$, where Δt_k^{ij} is the recorded past k^{th} meeting interval between u_i and u_j, and r_{ij} is the total number of recorded meeting intervals between u_i and u_j. The last contact between u_i and u_j occurred at time t_0^{ij}. Then, u_i can calculate its expected EV using Theorem 3.2.1.

Theorem 3.2.1. *At time t ($t \geq t_0^{ij}$), the expected encounter value of node u_i within $(t, t+\tau]$ is:*

$$EEV_i(t,\tau) = \sum_{1 \leq j \leq n, j \neq i} \frac{m_{ij}^{\tau}}{m_{ij}}, \tag{3.2.1}$$

where $M_{ij} = \{\Delta t_k^{ij} | \Delta t_k^{ij} \in R_{ij}, \Delta t_k^{ij} > t - t_0^{ij}\}$ and $m_{ij} = |M_{ij}|$, $M_{ij}^{\tau} = \{\Delta t_k^{ij} | \Delta t_k^{ij} \in M_{ij}, \Delta t_k^{ij} \leq t + \tau - t_0^{ij}\}$ and $m_{ij}^{\tau} = |M_{ij}^{\tau}|$.

Proof. Assume that the next meeting interval between u_i and u_j is Δt^{ij}, then the probability that u_i will meet u_j in $(t, t+\tau]$ is the probability that Δt^{ij} is not larger than $t_\tau - t_0^{ij}$ under the condition that $\Delta t^{ij} > t - t_0^{ij}$) (as u_i has not meet u_j since t_0^{ij}), which can be denoted as $P(\Delta t^{ij} \leq t + \tau - t_0^{ij} | \Delta t^{ij} > t - t_0^{ij})$. Thus,

$$EEV_i(t,\tau) = \sum_{1 \leq j \leq n,\ j \neq i} P(\Delta t^{ij} \leq t + \tau - t_0^{ij} | \Delta t^{ij} > t - t_0^{ij}).$$

Here,

$$P(\Delta t^{ij} \leq t + \tau - t_0^{ij} | \Delta t^{ij} > t - t_0^{ij}) = \frac{P(t - t_0^{ij} < \Delta t^{ij} \leq t + \tau - t_0^{ij})}{P(\Delta t^{ij} > t - t_0^{ij})}.$$

Considering $m_{ij} = |M_{ij}|$ where $M_{ij} = \{\Delta t_k^{ij} | \Delta t_k^{ij} \in R_{ij}, \Delta t_k^{ij} > t - t_0^{ij}\}$, and $m_{ij}^{\tau} = |M_{ij}^{\tau}|$ where $M_{ij}^{\tau} = \{\Delta t_k^{ij} | \Delta t_k^{ij} \in M_{ij}, \Delta t_k^{ij} \leq t + \tau - t_0^{ij}\}$, it can be obtained that:

$$P(\Delta t^{ij} > t - t_0^{ij}) = \sum_{\Delta t_k^{ij} \in M_{ij}} \frac{1}{r_{ij}} = \frac{m_{ij}}{r_{ij}},$$

and

$$P(t - t_0^{ij} < \Delta t^{ij} \leq t + \tau - t_0^{ij}) = \sum_{\Delta t_k^{ij} \in M_{ij}^{\tau}} \frac{1}{r_{ij}} = \frac{m_{ij}^{\tau}}{r_{ij}}.$$

So,

$$P(\Delta t^{ij} \leq t + \tau - t_0^{ij} | \Delta t^{ij} > t - t_0^{ij}) = \frac{m_{ij}^{\tau}/r_{ij}}{m_{ij}/r_{ij}} = \frac{m_{ij}^{\tau}}{m_{ij}}. \tag{3.2.2}$$

Therefore, it can be obtained that

$$EEV_i(t, \tau) = \sum_{1 \le j \le n, j \ne i} \frac{m_{ij}^{\tau}}{m_{ij}}.$$

□

According to Theorem 3.2.1, each node in the network can calculate its expected EV when it meets any other node.

3.2.1.2 Replicas Distribution

In the EER, each message is initiated with a predefined number of replicas in the network. Assume that the initial number of replicas of each message is λ. A message is considered to be successfully delivered if at least one replica arrives at the destination within the TTL of the message. Thus, to obtain a high message delivery ratio, an effective strategy is to disseminate the λ replicas of each message to λ different nodes firstly, and then let each of the λ different nodes deliver the single-copy message to the destination respectively.

As each node can calculate its expected EV in a fixed future time interval based on its contact history, when two nodes meet, the distribution of the replicas of each message can be conducted according to the proportion of their expected EVs. For example, if u_i holds a message m_k with M_k replicas ($M_k > 1$), and its current TTL is TTL$_k$, u_j has no replica of m_k. When u_i meets u_j at time t, after exchanging and updating the routing information, u_i will pass

$$\lfloor M_k \cdot \frac{EEV_j(t, \alpha \cdot TTL_k)}{EEV_i(t, \alpha \cdot TTL_k) + EEV_j(t, \alpha \cdot TTL_k)} \rfloor$$

replicas of message m_k to u_j, here α is a network parameter and $0 \le \alpha \le 1$. That is, u_i and u_j will distribute the M_k replicas of m_k according to the proportion of their expected EVs in $(t, t + \alpha \cdot TTL_k]$.

3.2.2 Single Replica Forwarding Phase

In the EER, each message initially has λ replicas. In the multiple replicas distribution phase, each node disseminates all the replicas of the message to different nodes as soon as possible. When the number of replicas of the message in one node reduces to 1, the single replica forwarding phase starts.

In the single replica forwarding phase, each node needs to decide whether or not to forward the message it holds to its current encounter. Previous research has shown that using the contact history can make the prediction of the meeting delays to other nodes with a high accuracy [30], which is useful in making a routing decision. Thus, each node can firstly take advantage of its contact history to predict the one-hop meeting delays to other nodes, and then estimate the multi-hop meeting delay, which is the minimum expected meeting delay (MEMD) to the destination. Finally, the node can decide whether to forward the message it holds to its current encounter by comparing their MEMDs to the destination.

3.2.2.1 One-Hop Meeting Delay Prediction

The one-hop meeting delay can be calculated based on the past contact information. In the previous work [30], the average meeting interval between two encounters is used as their expected meeting delay. For example, if node u_i has a set of recorded past meeting intervals $\{\Delta t_1^{ij}, \Delta t_2^{ij}, ..., \Delta t_{r_{ij}}^{ij}\}$ to node u_j. Then at any moment before u_i encounters u_j, it will predict the expected meeting delay to u_j as $\frac{1}{r_{ij}} \sum_{k=1}^{r_{ij}} \Delta t_k^{ij}$. However, this average meeting interval is not always appropriate to be the prediction of the meeting delay. For instance, if two nodes periodically meet every Δt, and the last moment these two nodes meet is t_0^{ij}, then at $t_0^{ij} + \frac{1}{2}\Delta t$, the expected meeting delay between these two nodes should be $\frac{1}{2}\Delta t$, but not the average meeting interval Δt. Thus, the elapsed time since last contact between two nodes does impact their expected meeting delay.

Theorem 3.2.2 can be used to calculate the expected meeting delay (EMD) between two nodes, which is related to the elapsed time since their last contact.

Theorem 3.2.2. *At time t ($t \geq t_0^{ij}$), the expected meeting delay (EMD) since t between nodes u_i and u_j is:*

$$EMD_{ij}(t) = \frac{1}{m_{ij}} \sum_{\Delta t_k^{ij} \in M_{ij}} \Delta t_k^{ij} - (t - t_0^{ij}), \tag{3.2.3}$$

where $M_{ij} = \{\Delta t_k^{ij} | \Delta t_k^{ij} \in R_{ij}, \Delta t_k^{ij} > t - t_0^{ij}\}$, and $m_{ij} = |M_{ij}|$.

Proof. Assume that the next meeting interval between nodes u_i and u_j is Δt^{ij}, then the expected meeting delay since t between $u)i$ and u_j is the expected value of $\Delta t^{ij} - (t - t_0^{ij})$ under the condition that $t - t_0^{ij}$, thus,

$$\begin{aligned} EMD_{ij}(t) &= E(\Delta t^{ij} - (t - t_0^{ij}) | \Delta t^{ij} > t - t_0^{ij}) \\ &= E(\Delta t^{ij} | \Delta t^{ij} > t - t_0^{ij}) - E(t - t_0^{ij} | \Delta t^{ij} > t - t_0^{ij}) \\ &= E(\Delta t^{ij} | \Delta t^{ij} > t - t_0^{ij}) - (t - t_0^{ij}). \end{aligned}$$

$E(\Delta t^{ij} | \Delta t^{ij} > t - t_0^{ij})$ can be calculated as:

$$\begin{aligned} E(\Delta t^{ij} | \Delta t^{ij} > t - t_0^{ij}) &= \sum_{k=1}^{r_{ij}} P(\Delta t^{ij} = \Delta t_k^{ij} | \Delta t^{ij} > t - t_0^{ij}) \cdot \Delta t_k^{ij} \\ &= \sum_{k=1}^{r_{ij}} \frac{P(\Delta t^{ij} = \Delta t_k^{ij}, \Delta t^{ij} > t - t_0^{ij})}{P(\Delta t^{ij} > t - t_0^{ij})} \cdot \Delta t_k^{ij}. \end{aligned}$$

Here,

$$P(\Delta t^{ij} = \Delta t_k^{ij}, \Delta t^{ij} > t - t_0^{ij}) = \begin{cases} \frac{1}{r_{ij}} & \text{if } \Delta t_k^{ij} > t - t_0^{ij}, \\ 0 & \text{else.} \end{cases}$$

and

$$P(\Delta t^{ij} > t - t_0^{ij}) = \sum_{\Delta t_k^{ij} \in M_{ij}} \frac{1}{r_{ij}} = \frac{m_{ij}}{r_{ij}}.$$

It can be obtained that:

$$E(\Delta t^{ij} | \Delta t^{ij} > t - t_0^{ij}) = \frac{r_{ij}}{m_{ij}} \sum_{\Delta t_k^{ij} \in M_{ij}} \frac{1}{r_{ij}} \cdot \Delta t_k^{ij} = \frac{1}{m_{ij}} \sum_{\Delta t_k^{ij} \in M_{ij}} \Delta t_k^{ij}.$$

Therefore,

$$\begin{aligned} EMD_{ij}(t) &= E(\Delta t^{ij} | \Delta t^{ij} > t - t_0^{ij}) - (t - t_0^{ij}) \\ &= \frac{1}{m_{ij}} \sum_{\Delta t_k^{ij} \in M_{ij}} \Delta t_k^{ij} - (t - t_0^{ij}). \end{aligned}$$

□

According to Eq. (3.2.3), each node in the network can predict the one-hop meeting delays between itself and other nodes. However, the one-hop meeting delay only includes partial connectivity information of the network. The global network connectivity information is more useful for the message delivery in the single replica forwarding.

3.2.2.2 Multi-Hop Meeting Delay Prediction

To make the message efficiently delivered to its destination in a DTN, each node can estimate the multi-hop meeting delay from itself to the destination, which is used to determine whether to forward the message to the current encounter. Before calculating the multi-hop meeting delay, each node can make its one-hop meeting delay prediction and exchange it with other encounters, through which it can get the network connectivity information. In the EER, each node maintains an $n \times n$ meeting interval matrix MI. The MI is defined as:

$$\mathbf{MI} = \begin{pmatrix} 0 & I_{12} & \dots & I_{1n} \\ I_{21} & 0 & \dots & I_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ I_{n1} & I_{n2} & \dots & 0 \end{pmatrix},$$

where I_{ij} denotes the average meeting interval between nodes u_i and u_j and it is updated by u_i. Obviously, $I_{ij} = 0$ when $i = j$. For the *MI* of u_i, I_{ij} $(1 \leq j \leq n, j \neq i)$ can be obtained by $I_{ij} = \frac{1}{r_{ij}} \sum_{k=1}^{r_{ij}} \Delta t_k^{ij}$. The other elements in the *MI* of u_i can be obtained via information exchange when it meets other nodes. For the convenience of exchanging the meeting interval information when two nodes meet, each node has to maintain the last update time for each row in its *MI*. When two nodes meet at another time, they will exchange and update their *MI*s with each other according to the last update time of each row.[3.2]

As mentioned above, the prediction of the meeting delay based on the average meeting intervals may not be always accurate. Each node will build an $n \times n$ expected meeting delay matrix *MD* whenever it meets another node, where

$$\mathbf{MD} = \begin{pmatrix} 0 & D_{12} & \dots & D_{1n} \\ D_{21} & 0 & \dots & D_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{n1} & D_{n2} & \dots & 0 \end{pmatrix}.$$

D_{ij} is the expected meeting delay (EMD) between nodes u_i and u_j, which is updated by u_i. Also, $D_{ij} = 0$ when $i = j$. In the *MD* of u_i, D_{ij} $(1 \leq j \leq n, j \neq i)$ can be obtained by Eq. (3.2.3). As it is difficult for u_i to get the EMD between u_j and u_k when $j \neq i$ and $k \neq i$, u_i can replace it with I_{jk} for simplicity, which can be acquired from its *MI*. After building the *MD*, u_i can calculate the multi-hop meeting delay from itself to the destination of the message using the Dijkstra's algorithm.

Theorem 3.2.3. *The multi-hop meeting delay calculated above is the minimum expected meeting delay (MEMD).*

Proof. As each element in MD indicates the EMD between a pair of nodes, it is easy to see

[3.2] In the implementation of the protocols, only the rows with the fresher update time need to be exchanged between the two encountering nodes, which can reduce the routing information exchange overhead greatly.

that the calculated multi-hop meeting delay from the node to a particular destination using the Dijkstra's algorithm is the MEMD between the node and the destination. □

3.2.3 Expected Encounter Based Routing Algorithm

Algorithm 1 Expected Encounter Based Routing Algorithm

```
Let $m_1, m_2, \dots, m_M$ be the messages in $u_i$'s local buffer.
if $u_i$ meets $u_j$ at $t$ then
    $u_i$ and $u_j$ update their contact histories and calculate the up-to-date average meeting interval.
    $u_i$ and $u_j$ exchange their $MIs$ with each other to form an identical $MI$.
    $u_i$ and $u_j$ build their $MDs$.
    for $k = 1, 2, \dots, M$ do
        if $u_j$ does not hold $m_k$ then
            $M_k \leftarrow m_k.numOfReplicas$
            if $M_k > 1$ then
                $u_i$ sends $\lfloor M_k \cdot \frac{EEV_j(t, \alpha \cdot TTL_k)}{EEV_i(t, \alpha \cdot TTL_k) + EEV_j(t, \alpha \cdot TTL_k)} \rfloor$ replicas of $m_k$ to $u_j$.
            else
                $u_d \leftarrow m_k.destination$
                if $MEMD(u_i, u_d) > MEMD(u_j, u_d)$ then
                    $u_i$ forwards $m_k$ to $u_j$.
                end if
            end if
        end if
    end for
end if
```

The procedure of the expected encounter based routing algorithm is described in Algorithm 1. When nodes u_i and u_j meet, they update their contact histories for the meeting intervals between them and calculate the up-to-date average meeting interval. Then u_i and u_j exchange their MIs with each other to form an identical MI. If either one of these two nodes has a message to be delivered, each of them will build a new MD based on its MI and the one-hop meeting delay prediction. For each message m_k, which is held by u_i but not u_j, if u_i has M_k replicas of m_k ($M_k > 1$), it will send $\lfloor M_k \cdot \frac{EEV_j(t, \alpha \cdot TTL_k)}{EEV_i(t, \alpha \cdot TTL_k) + EEV_j(t, \alpha \cdot TTL_k)} \rfloor$ replicas of m_k to u_j, and keep the rest replicas of m_k. Otherwise, if u_i has only one replica of m_k, it compares its

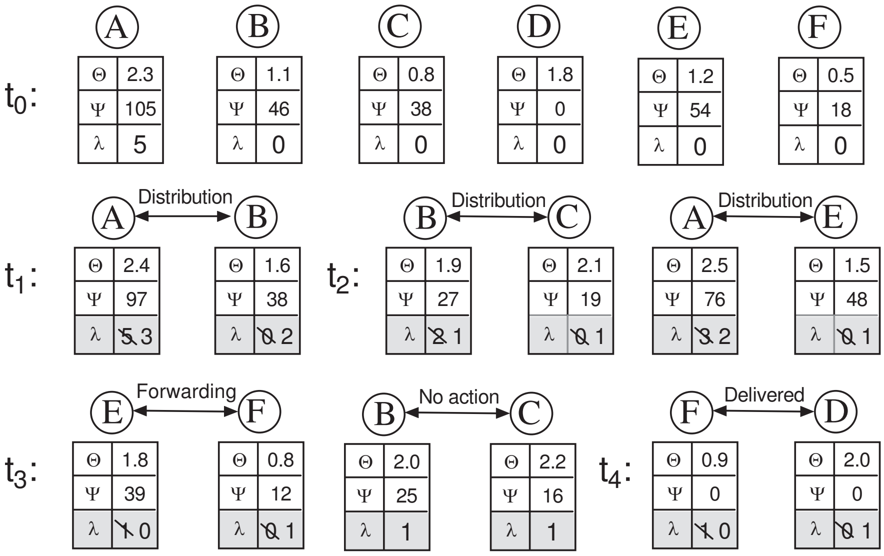


Figure 3.2. Procedure of the EER. Message m_k is generated by node A, and its destination is node D. Θ denotes the expected EV, Ψ denotes the MEMD and λ denotes the number of replicas of m_k.

MEMD to the destination with that of u_j, i.e., it compares $MEMD(u_i, u_d)$ with $MEMD(u_j, u_d)$ where $MEMD(u_*, u_d)$ denotes the minimum expected meeting delay from u_* to u_d. If u_i has a longer MEMD, it will forward m_k to u_j. Note that the replicas of each message will not be redistributed between two encounters if both of them have at least one replica of this message.

Example 1. Figure 3.2 illustrates the procedure of the EER for the sample DTN given in Figure 3.1. Initially, node A generates a message m_k with 5 replicas at time t_0. m_k's destination is node D. At t_1, node A meets node B, the expected EVs of node A and B are 2.4 and 1.6 respectively, so node A sends $\lfloor 5 \cdot \frac{1.6}{2.4+1.6} \rfloor = 2$ replicas of m_k to node B. Thus, node A updates its number of replicas to 3, and node B has 2 replicas of m_k. Similarly, at t_2, node B sends $\lfloor 2 \cdot \frac{2.1}{1.9+2.1} \rfloor = 1$ replica of m_k to node C, and node A sends $\lfloor 3 \cdot \frac{1.5}{2.5+1.5} \rfloor = 1$ replica of m_k to node E. At t_3, node E meets node F. As node E has only 1 replica of m_k, it will decide whether to forward m_k to node F by comparing their MEMDs. The MEMDs of nodes E and F are 39 and 12 respectively. Node E will forward m_k to node F since the latter has a lower MEMD than that of node E. At the same time, node B meets node C and there will be no action between them since each of them has one replica of m_k. Finally, node F meets node D at t_4, and m_k is then successfully delivered to node D.

3.3 Community Based Routing Protocol

In the EER, each node maintains an MI, which includes the global network connectivity information. When a pair of nodes meet, they will exchange and update their MIs, which may cause some routing information exchange overhead. This section will propose the community based routing protocol (CR), which employs the concept of social network and can further reduce the information exchange overhead by diminishing the scale of the MIs. Firstly, the concept of community will be introduced. Then the calculation of expected number of encountering communities for each node in a fixed future time interval will be described. Finally, the community based routing algorithm will be elaborated.

3.3.1 Community — A Social Network Concept

A social network is a structured human society, which consists of individuals (called nodes) connected by socially meaningful relationships, such as common interest or social relations. Such social relationships can also partition the social network into several communities naturally. Community is an important attribute of a social network. Generally, the social relationship within the same community is stronger than that between different communities. For instance, the contact frequency between a pair of nodes in the same community is much higher than that from different communities. More specifically, for example, all the students in a school are divided into different classes (i.e., communities). The students from the same class will meet with each other frequently as they are classmates and they attend similar classes together. On the other hand, the meeting frequency between the students from different classes will be much lower.

In this chapter, we define a community as: a community contains some nodes which have high trace overlapping and each of them will encounter each other more frequently. The concept of community can be used in the DTN routing. In a DTN, all the nodes are divided into several communities according to their relationships. Then the routing in the DTN can be conducted in two phases — inter-community routing and intra-community routing. In the inter-community routing, each node distributes the multiple replicas of a message to the nodes from different communities as soon as possible, which can be achieved by distributing the replicas of the message according to the proportion of the two encounters' expected numbers of encountering communities. In case of the single replica of the message, it will be forwarded to the node, which has a higher probability to encounter the destination community (i.e., the community, which the destination of the message belongs to). In the intra-community routing, a node in the destination community distributes the replicas of a message to its encounter in the same community according to the proportion of their intra-community expected EVs, which are calculated based on the nodes only in the same community. Note that the intra-community MEMD, intra-community MI and intra-community MD that will be discussed in the following sections are also calculated based on the nodes only in the same community. In case of the single replica of the message, the node in the destination community decides whether to forward the message to its encounter in the same community by comparing their intra-community MEMDs.

There are a lot of research work on the construction of community, including the centralized algorithms, such as the *k*-clique [67] and weighted network analysis (WNA) [60], and distributed algorithms such as the construction method in [11]. However, the construction of community is not the focus in this chapter. This chapter will take advantages of the community property and propose the community based routing protocol (CR).[3.3]

[3.3] In the implementation of the CR, the communities in the network are predefined for simplicity.

3.3.2 Expected Number of Encountering Communities

In a community based DTN, each node maintains the intra-community MI and MD. In addition, each node also needs to maintain $n-1$ sliding windows to record contact histories, i.e., the past meeting intervals between itself and any other $n-1$ nodes. The node can use the recorded contract histories to calculate its expected number of encountering communities in a fixed future time interval.

In this chapter, it is assumed that a network is partitioned into l communities $\{C_1, C_2, \cdots, C_l\}$ and each node only belongs to one of the l communities.[3.4] C_k denotes the set of nodes inside the k^{th} community, and CID_{u_i} denotes the ID of the community, which node u_i belongs to. u_i is considered to encounter community C_k if it meets at least one node in C_k. Then u_i can calculate its expected number of encountering communities using Theorem 3.3.1.

Theorem 3.3.1. *At time t ($t \geq t_0^{ij}$), the expected number of encountering communities for node u_i within $(t, t+\tau]$ is:*

$$ENEC_i(t,\tau) = \sum_{1\leq k\leq l, k\neq CID_{u_i}} (1 - \prod_{u_j \in C_k} (1 - \frac{m_{ij}^{\tau}}{m_{ij}})), \tag{3.3.1}$$

where $M_{ij} = \{\Delta t_k^{ij} | \Delta t_k^{ij} \in R_{ij}, \Delta t_k^{ij} > t - t_0^{ij}\}$ and $m_{ij} = |M_{ij}|$, $M_{ij}^{\tau} = \{\Delta t_k^{ij} | \Delta t_k^{ij} \in M_{ij}, \Delta t_k^{ij} \leq t + \tau - t_0^{ij}\}$ and $m_{ij}^{\tau} = |M_{ij}^{\tau}|$.

Proof. Assume that the next meeting interval between nodes u_i and u_j is Δt^{ij}, and the probability that u_i will encounter community C_k in $(t, t+\tau]$ is P_{ik}. The expected number of encountering communities for u_i in $(t, t+\tau]$ is the summation of the probabilities that u_i will encounter each of the communities, thus:

[3.4] This chapter only considers that one node belongs to one community for simplicity of description. It is noted that the proposed protocol can work well even when one node belongs to multiple communities.

$$ENEC_i(t,\tau) = \sum_{1\leq k\leq l, k\neq CID_{u_i}} P_{ik}.$$

P_{ik} can be calculated as:

$$P_{ik} = 1 - \prod_{u_j\in C_k}(1 - P(\Delta t^{ij} \leq t + \tau - t_0^{ij}|\Delta t^{ij} > t - t_0^{ij})).$$

In Eq. (3.2.2) of Theorem 3.2.1, it has already been obtained that:

$$P(\Delta t^{ij} \leq t + \tau - t_0^{ij}|\Delta t^{ij} > t - t_0^{ij}) = \frac{m_{ij}^{\tau}}{m_{ij}},$$

where $M_{ij} = \{\Delta t_k^{ij}|\Delta t_k^{ij} \in R_{ij}, \Delta t_k^{ij} > t - t_0^{ij}\}$ and $m_{ij} = |M_{ij}|$, $M_{ij}^{\tau} = \{\Delta t_k^{ij}|\Delta t_k^{ij} \in M_{ij}, \Delta t_k^{ij} \leq t + \tau - t_0^{ij}\}$ and $m_{ij}^{\tau} = |M_{ij}^{\tau}|$.

Thus,

$$ENEC_i(t,\tau) = \sum_{1\leq k\leq l, k\neq CID_{u_i}} P_{ik} = \sum_{1\leq k\leq l, k\neq CID_{u_i}} (1 - \prod_{u_j\in C_k}(1 - \frac{m_{ij}^{\tau}}{m_{ij}})).$$

□

Based on Theorem 3.3.1, each node in the network can calculate its expected number of encountering communities when it meets any other node.

3.3.3 Community Based Routing Algorithm

In the community based routing protocol, every node has a global unique ID and a community ID. When a message is generated, its destination u_d is attached in this message together with the community ID CID_{u_d}.

The community based routing algorithm is shown in Algorithm 2. When nodes u_i and u_j meet, they update their contact histories and calculate the up-to-date average meeting interval. For each message m_k, which is held by u_i, if the destination of m_k is not within the same

Algorithm 2 Community Based Routing Algorithm

```
Let $m_1, m_2, \ldots, m_M$ be the messages in $u_i$'s local buffer.
if $u_i$ meets $u_j$ at $t$ then
    $u_i$ and $u_j$ update their contact histories and calculate the up-to-date average meeting interval.
    for $k = 1, 2, \ldots, M$ do
        $u_d \leftarrow m_k.destination$
        if $CID_{u_i} \neq CID_{u_d}$ then
            Trigger Inter-Community Routing Algorithm.
        else
            Trigger Intra-Community Routing Algorithm.
        end if
    end for
end if
```

Algorithm 3 Inter-Community Routing Algorithm

```
if $CID_{u_j} = CID_{u_d}$ then
    $u_i$ sends all replicas of $m_k$ to $u_j$.
else
    if $u_j$ does not hold $m_k$ then
        $M_k \leftarrow m_k.numOfReplicas$
        if $M_k > 1$ then
            $u_i$ sends $\lfloor M_k \cdot \frac{ENEC_j(t,\alpha \cdot TTL_k)}{ENEC_i(t,\alpha \cdot TTL_k)+ENEC_j(t,\alpha \cdot TTL_k)} \rfloor$ replicas of $m_k$ to $u_j$.
        else
            $c \leftarrow CID_{u_d}$
            if $P_{ic} < P_{jc}$ then
                $u_i$ forwards $m_k$ to $u_j$.
            end if
        end if
    end if
end if
```

community of u_i, the inter-community routing algorithm will be triggered. Otherwise, the intra-community routing algorithm will be triggered.

The inter-community and intra-community routing algorithms are described in Algorithm 3 and Algorithm 4 respectively. In the inter-community routing algorithm, u_i tries to deliver m_k to the destination community. If u_j belongs to the destination community, u_i will send all the replicas of m_k to u_j. Otherwise, u_i will continue the process: If m_k is held by u_i but not u_j, u_i will

Algorithm 4 Intra-Community Routing Algorithm

```
if $CID_{u_i} = CID_{u_j}$ then
    $u_i$ and $u_j$ exchange their intra-community $MIs$ with each other to form an identical intra-community $MI$.
    $u_i$ and $u_j$ build their intra-community $MDs$.
    if $u_j$ does not hold $m_k$ then
        $M_k \leftarrow m_k.numOfReplicas$
        if $M_k > 1$ then
            $u_i$ sends $\lfloor M_k \cdot \frac{EEV_j(t,\alpha \cdot TTL_k)'}{EEV_i(t,\alpha \cdot TTL_k)' + EEV_j(t,\alpha \cdot TTL_k)'} \rfloor$ replicas of $m_k$ to $u_j$.
        else
            if $MEMD(u_i, u_d)' > MEMD(u_j, u_d)'$ then
                $u_i$ forwards $m_k$ to $u_j$.
            end if
        end if
    end if
end if
```

make the routing decision according to M_k (the number of replicas of m_k). If M_k is larger than 1, u_i will distribute the replicas of m_k between itself and u_j according to the proportion of their expected numbers of encountering communities. u_i will send $\lfloor M_k \cdot \frac{ENEC_j(t,\alpha \cdot TTL_k)}{ENEC_i(t,\alpha \cdot TTL_k) + ENEC_j(t,\alpha \cdot TTL_k)} \rfloor$ replicas of m_k to u_j. Otherwise, if u_i has only one replica of m_k, and if the probability that u_i will encounter the destination community in $(t, t + \alpha \cdot TTL_k]$ is less than that of u_j, u_i will forward m_k to u_j.

In the intra-community routing algorithm, if u_i and u_j belong to different communities, u_i will not send m_k to u_j as u_j is outside the destination community. Otherwise, nodes u_i and u_j will exchange their intra-community MIs to form an identical intra-community MI and build their new intra-community MDs. If m_k is held by node u_i but not u_j, u_i will make the routing decision according to the number of replicas of m_k. If u_i has more than one replica of m_k, it will send $\lfloor M_k \cdot \frac{EEV_j(t,\alpha \cdot TTL_k)'}{EEV_i(t,\alpha \cdot TTL_k)' + EEV_j(t,\alpha \cdot TTL_k)'} \rfloor$ replicas of m_k to u_j, where $EEV_i(t, \alpha \cdot TTL_k)'$ represents the intra-community expected EV of u_i in $(t, t + \alpha \cdot TTL_k]$. If u_i has only one replica of m_k and the $MEMD'$ (intra-community $MEMD$) from u_i to u_d is larger than that from u_j to u_d, u_i will forward m_k to u_j.

Example 2. Figure 3.3 illustrates the procedure of the CR for the sample DTN in Figure 3.1. Node A belongs to community C_1, nodes B and C belong to community C_2, nodes D, E and F belong to community C_3. Initially, node A generates a message m_k with 5 replicas at time t_0. m_k's destination is node D thus the destination community is C_3. At t_1, node A meets node B, their expected numbers of encountering communities are 2.7 and 0.9 respectively, so node A sends $\lfloor 5 \cdot \frac{0.9}{2.7+0.9} \rfloor = 1$ replica of m_k to node B. Thus, node A updates its number of replicas of m_k to 4. Similarly, at t_2, node B meets node C. As node B has only 1 replica of m_k, it will decide whether to forward m_k to node C by comparing their probabilities to encounter the destination community. Node B will forward m_k to node C since the latter has a higher probability to encounter the destination community. At the same time, node A meets node E and node A sends all the replicas of m_k to node E since the latter is in the destination community. At t_3, node E meets node F, their intra-community expected EVs are 2.8 and 1.2 respectively, so node E sends $\lfloor 4 \cdot \frac{1.2}{2.8+1.2} \rfloor = 1$ replica of m_k to node F. At the same time, node B meets node C. As node C has a higher probability to encounter the destination community, it will keep the replica of m_k. Finally, node F meets node D at t_4, and m_k is then successfully delivered to node D.

3.4 Performance Evaluation

In this section, the proposed routing protocols will be evaluated with three performance metrics, delivery ratio, latency and goodput, in the Opportunistic Network Environment simulator (ONE) [33]. The comparison between the proposed routing protocols and other existing popular DTN routing protocols will also be conducted.

3.4.1 Performance Metrics and Simulation Settings

Three metrics will be employed in the performance evaluation, including delivery ratio, latency and goodput. The major goal of the DTN routing is to achieve a high delivery ratio and goodput

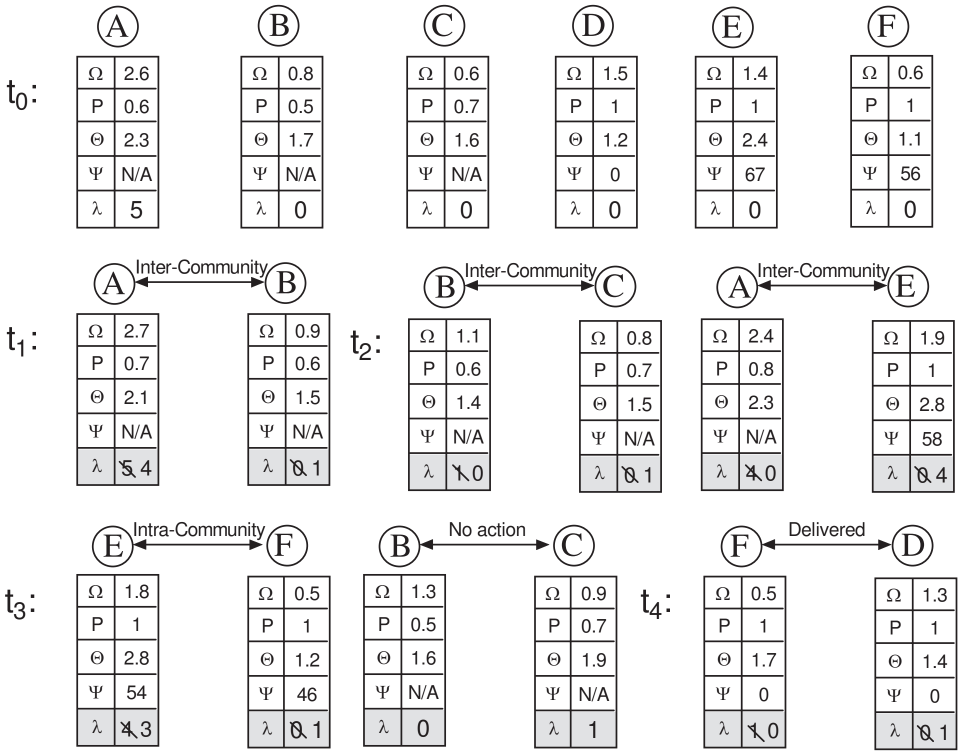


Figure 3.3. Procedure of the CR. Message m_k is generated by node A, and its destination is node D. Ω denotes the expected number of encountering communities and P is the probability to encounter the destination community. Θ denotes the intra-community expected EV, Ψ denotes the intra-community minimum expected meeting delay ($MEMD'$) and λ denotes the number of replicas of m_k.

with a low latency. The definitions of these three metrics are shown as follows:

- *Delivery ratio:* The ratio of the number of delivered messages to the number of all the generated messages.

- *Latency:* The average end-to-end delivery delay between each pair of source and destination in the network.

- *Goodput:* The ratio of the number of delivered messages to the total number of relayed messages in the network.

To evaluate the performance of the proposed DTN routing protocols, the vehicular-based map-driven model is used in the simulations, which is part of the ONE simulator. Some bus lines based on an approximate 5 km x 3 km section of downtown Helsinki in Finland are employed into the simulations, and the buses, which travel along the bus lines, represent the nodes in the network.

The settings in the simulations are as follows: the moving speed of the nodes varies from 2.7 to 13.9 m/s, the simulation update interval is $0.1s$, the transmission speed is 2 Mbps and the transmission range is 10 m. The buffer space of each node is 1 MB, and the size of each packet is 25 KB. The message generation rate is 2 per minutes. The network parameter α is set to 0.28, which is indicated to be a reasonable value from the preliminary simulations. Each simulation lasts for $10000s$, and the TTL of each message is 20 minutes. The number of nodes in the network varies from 40 to 240 with an increment of 40. The value of each point in the curves is the average of 10 simulation runs. For the CR protocol, the buses are manually divided into 10 communities based on the line overlapping of all the buses and each bus only belongs to one of the communities.

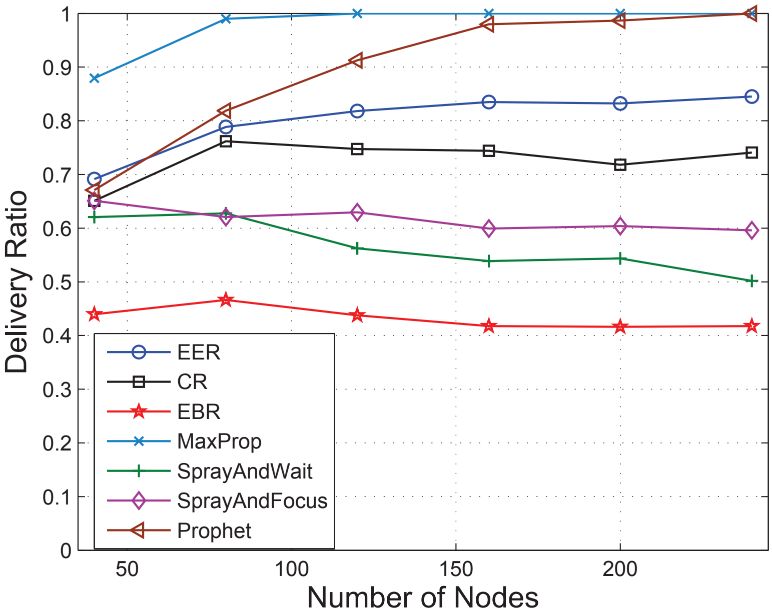


Figure 3.4. Delivery ratio comparison between the proposed routing protocols and other existing protocols.

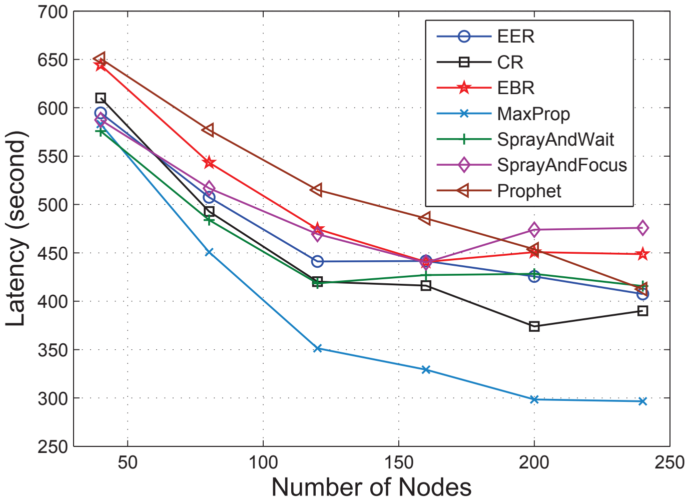


Figure 3.5. Latency comparison between the proposed routing protocols and other existing protocols.

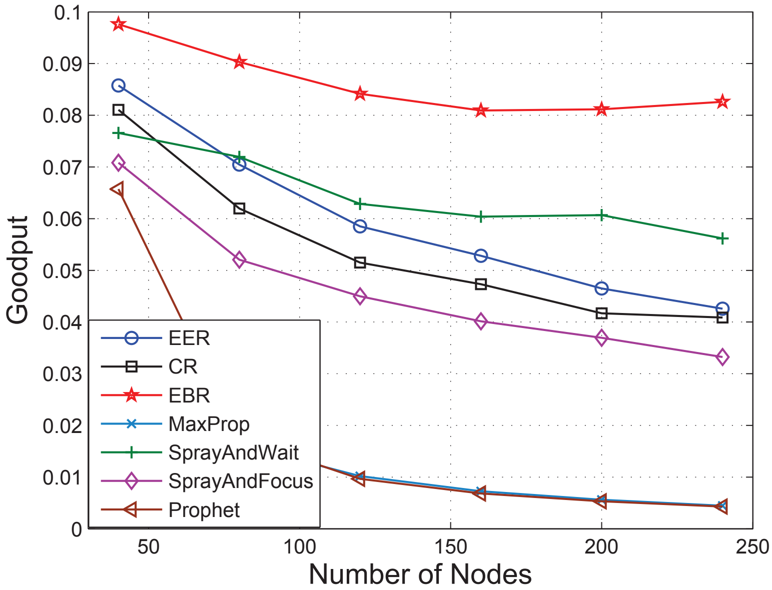


Figure 3.6. Goodput comparison between the proposed routing protocols and other existing protocols.

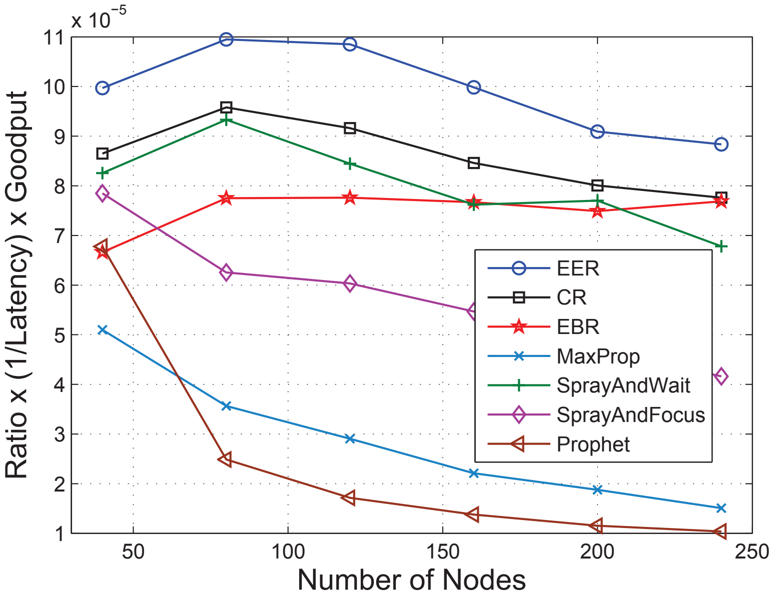


Figure 3.7. Composite performance comparison between the proposed routing protocols and other existing protocols.

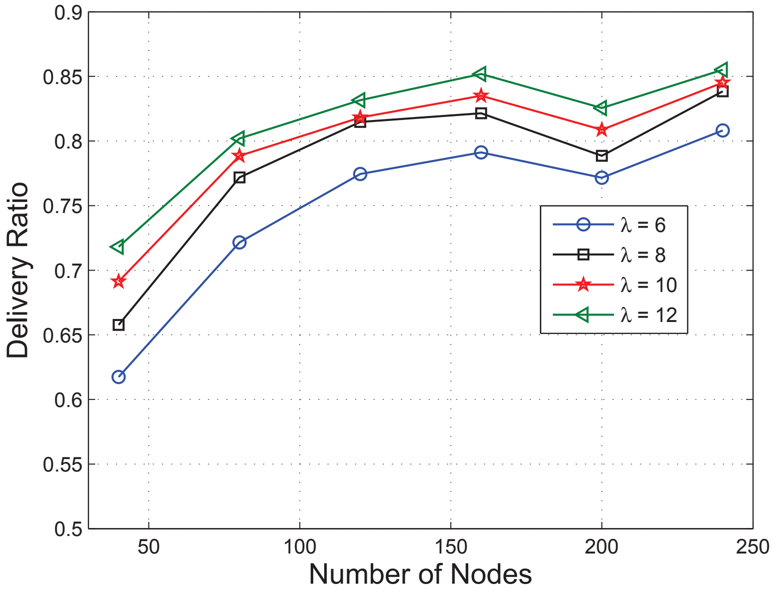


Figure 3.8. Effects of the value of λ on delivery ratio of the EER.

3.4.2 Simulation Results

To evaluate the effectiveness of the proposed DTN routing protocols, the EER and CR protocols are compared with other five popular protocols: EBR [57], MaxProp [5], Spray-and-Wait [78], Spray-and-Focus [79] and Prophet [46]. After that the effects of λ, which is the initial value of replicas of a message, on the performance of the EER and CR are analyzed respectively.

Figure 3.4, Figure 3.5 and Figure 3.6 show the performance comparison between the EER, CR and other five protocols. λ are set as 10 for the EER, CR, Spray-and-Wait and Spray-and-Focus protocols. The figure shows that MaxProp achieves the highest delivery ratio, shortest latency with the lowest goodput since it is an epidemic-based protocol. As another epidemic-based protocol, Prophet achieves a high delivery ratio, while its goodput is also very low. Thus, the MaxProp and Prophet fail in the comparison due to their poor goodput. The EBR obtains the best goodput, but its delivery ratio is the lowest and its latency is just a bit better than Prophet. The goodput of Spray-and-Wait exceeds those of EER and CR when the number of

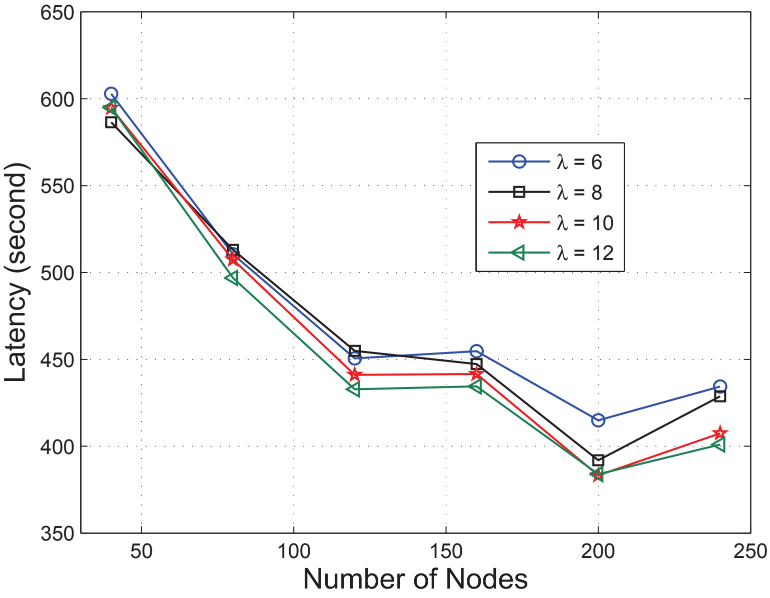


Figure 3.9. Effects of the value of λ on latency of the EER.

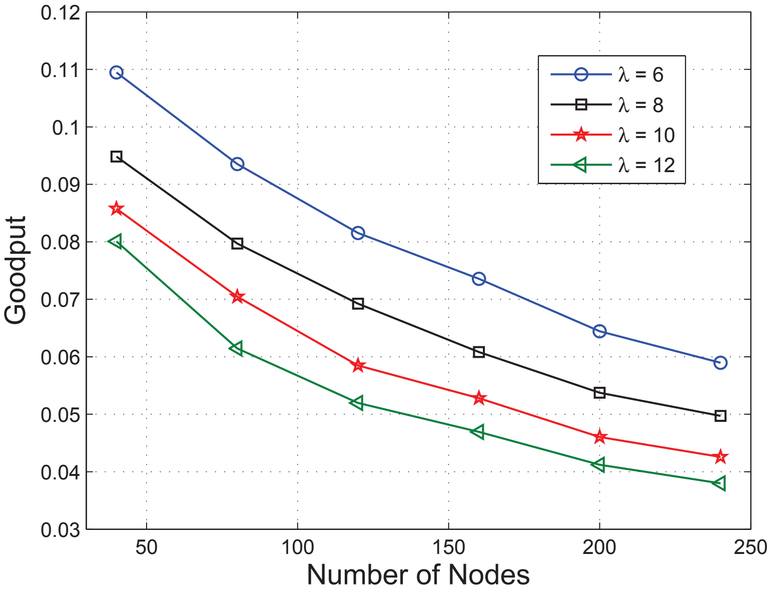


Figure 3.10. Effects of the value of λ on goodput of the EER.

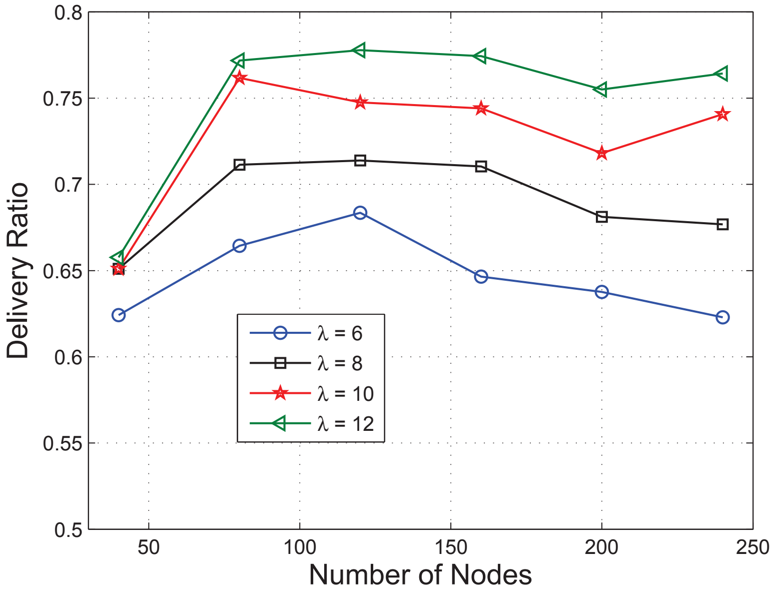


Figure 3.11. Effects of the value of λ on delivery ratio of the CR.

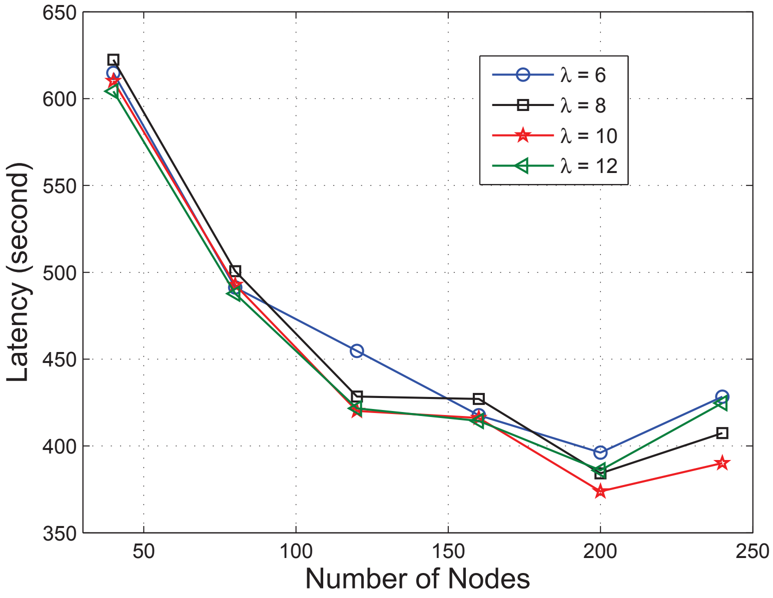


Figure 3.12. Effects of the value of λ on latency of the CR.

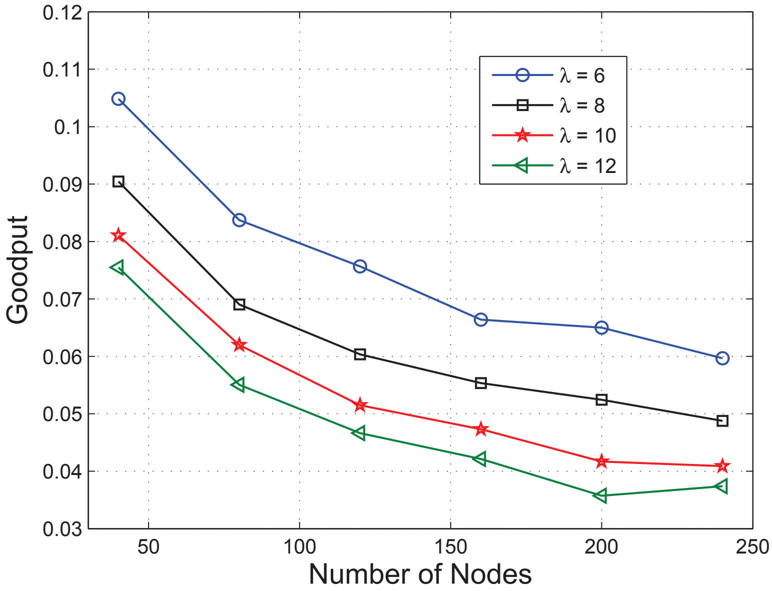


Figure 3.13. Effects of the value of λ on goodput of the CR.

nodes is larger than 80. However, its delivery ratio is much lower than EER and CR, and its latency is comparative to EER and CR. The Spray-and-Focus acquires a lower delivery ratio than EER and CR, with a higher latency and a lower goodput. Consequently, the proposed protocols EER and CR perform effectively compared with other five protocols.

Figure 3.7 shows the composite performance comparison between the proposed protocols and other five protocols. A composite metric is proposed in [57] to evaluate the overall performance of a specific protocol. The composite metric is defined as $Ratio \times (1/Latency) \times Goodput$, which encourages the one with a high message delivery ratio and high goodput and penalizes it with a long end-to-end delivery delay. It can be found from the figure that the proposed protocol EER performs the best followed by the CR, which both outperform other protocols.

Figure 3.8, Figure 3.9, Figure 3.10, Figure 3.11, Figure 3.12 and Figure 3.13 illustrate the effects of λ on the performance of EER and CR. The value of λ varies from 6 to 12 with an increment of 2. The delivery ratio of both protocols rises when the value of λ increases. The increase of λ can slightly reduce the latency (obvious in Figure 3.9). However, the increase of λ can heighten the overhead because a larger number of forwards will be employed in the network, and the EER and CR will achieve a lower goodput. Therefore, it is a tradeoff to determine an appropriate value of λ.

3.5 Summary

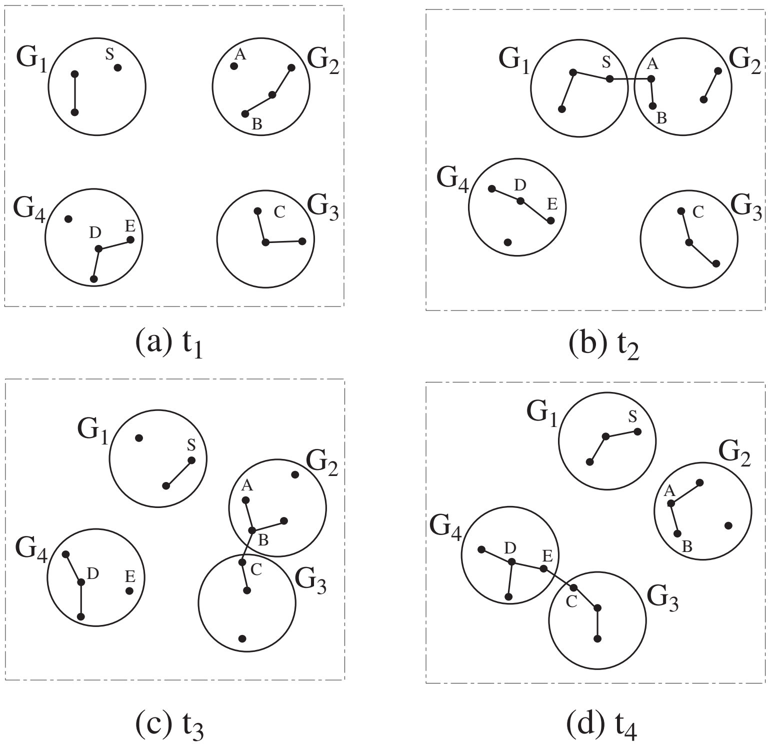
This chapter first proposes an expected encounter based routing protocol (EER), which distributes multiple replicas of a message proportionally between two encounters according to their expected EVs. In case of single replica of a message, EER makes the routing decision by comparing the minimum expected meeting delay to the destination. To take advantages of the community property, a community based routing protocol (CR), which is divided into inter-community routing and intra-community routing, is further proposed. The proposed routing protocols are evaluated in the ONE simulator under different parameters to demonstrate their effectiveness.

CHAPTER 4
GROUP AWARE COOPERATIVE ROUTING PROTOCOL UNDER RESOURCE CONSTRAINTS

4.1 Introduction

In Delay Tolerant Networks (DTNs), the paths between the source and destination are intermittently connected. Thus the conventional routing protocols [71, 29, 70] are not applicable and the store-carry-and-forward mechanism [87] is always adopted to deliver message in this kind of networks. In most of DTN routing protocols, each node determines the message transfer sequence independently when exchanging messages with its encounter. However, due to sporadic node density and unpredictable node mobility, the contact duration between each pair of nodes may be not long enough for them to transfer all the messages. Therefore, it is necessary for two encountering nodes to cooperate with each other to negotiate the message transfer sequence to efficiently make use of the limited contact duration [53].

DTNs can be formed by mobile nodes such as the portable devices carried by human beings [69]. In such DTNs, mobile nodes with common interest or close relationship tend to form into groups and move together. One typical DTN routing application scenario is the disaster recovery system [58]. For example, an intense earthquake may devastate most infrastructures of cellular networks, making the communication services broken down for a long time. In many post-earthquake scenarios, survivors would tend to move in groups to assist each other in case of further disasters. Thus, the mobile phones carried by moving people can construct a DTN with group feature to provide emergent communication services. Under such emergent situations, the survivals would likely try to contact their relatives or friends as early as



(a) t$_1$ (b) t$_2$

(c) t$_3$ (d) t$_4$

Figure 4.1. Message delivery in a sample delay tolerant network with group feature. G_1, G_2, G_3 and G_4 are four groups in the network.

possible, which will bring about a huge communication workload for the network and lead to the stringent constraints of the bandwidth and buffer space for the mobile nodes. This challenging scenario motivates us to design a group aware routing protocol for the DTNs under resource constraints. As the nodes within a group likely encounter each other more frequently and the connectivity among the group members is relatively stable, the source node can apply the strategy by first delivering the message to the destination group, and then letting the message be routed to the destination using the intra-group links. Moreover, the nodes can also cooperate with their group members to share their limited buffers in caching the messages more efficiently.

Figure 4.1 shows an example of message delivery in the DTNs, which is formed by four groups of nodes, G_1, G_2, G_3 and G_4. Suppose that the source node S, which is from group G_1, wants to deliver a message m_k to the destination node D, which is inside group G_4, at time t_1. The network can adopt the following routing procedure: When node S encounters node A at time t_2, node S can forward m_k to node A. At time t_3, node B meets node C. As node C has a higher probability than node B to meet the destination group G_4, node B can firstly request m_k from node A and then transfer it to node C. At time t_4, node C encounters node E which is in the destination group, then it can transfer m_k to node E. Finally, node E can deliver m_k to node D as they are in the same group. Thus, by using the group feature of the mobile nodes and further estimating the contact probabilities of diffident groups, the message can be efficiently delivered in the opportunistic networks.

This chapter proposes a cooperative routing protocol using the group feature of mobile nodes, which aims to maximize the message delivery probability in the DTN under resource constraints of both the bandwidth and buffer space. The proposed routing protocol includes a cooperative message transfer scheme and a buffer management strategy. In the cooperative message transfer scheme, the limited bandwidth available for mobile nodes is considered and two encountering nodes will exchange messages cooperatively to maximize the delivery probability. In the buffer management strategy, the constraint of mobile nodes' buffer space is further considered, and the cooperative message caching scheme, in which the message dropping priorities are designed to minimize the reduced delivery probability, is proposed.

4.2 System Model

A delay tolerant network with n nodes moving across a rectangular region is considered in this chapter. We employ the concept of group into this chapter and each group contains some nodes which will move with the same direction and the geographical relationships among the nodes within the same group are close and stable. The nodes are partitioned into m groups. Node i is denoted as v_i and group j is denoted as G_j. For a message m_k, the group, which the destination node of m_k belongs to, is defined as the *destination group*. Within each group, the nodes have a close geographical relationship and their connections are relatively frequent compared with those among the nodes from different groups.

The transmission range of each node is R. Two nodes v_i and v_j can communicate with each other at time t only if they are within the transmission range of each other, i.e., $|X_i(t)-X_j(t)| \leq R$, where $X_i(t)$ and $X_j(t)$ represent the positions of v_i and v_j at time t respectively. For the case that a node has more than one neighbor within its transmission range, it will randomly choose one of them to exchange the messages. The s^{th} *meeting time* between nodes v_i and v_j, denoted as $t_m^{ij}(s)$, is defined as the time they first come within the transmission range of each other since their $(s-1)^{th}$ leaving time. Thus, $t_m^{ij}(s) = min\{t|t > t_l^{ij}(s-1), |X_i(t) - X_j(t)| \leq R\}$, where $t_l^{ij}(s)$ is the s^{th} *leaving time* between nodes v_i and v_j presenting the time they first depart outside the transmission range of each other since their s^{th} meeting time $t_m^{ij}(s)$. Thus, $t_l^{ij}(s) = min\{t|t > t_m^{ij}(s), |X_i(t) - X_j(t)| > R\}$. Initially, if nodes v_i and v_j are connected at time t_0, then $t_m^{ij}(0) = t_0$ and $t_l^{ij}(0) = min\{t|t > t_0, |X_i(t) - X_j(t)| > R\}$. Otherwise, $t_m^{ij}(0) = t_l^{ij}(0) = t_0$. Considering node v_i holds a message with S_m KB and the transmission rate is T_r KBps, then node v_i can successfully transmit the message to node v_j at time $t_m^{ij}(s)$ only if $t_l^{ij}(s) - t_m^{ij}(s) \geq S_m/T_r$. Each generated message has a time-to-live (TTL)[4.1], within which it has to be delivered to the destination, otherwise, it will expire. As the buffer size S_b of each node is limited, it is necessary to propose an efficient buffer management strategy to schedule the message dropping in case that the buffer is full during the routing procedure.

4.3 Group Aware Cooperative Routing Protocol

In DTNs, the contact duration between each pair of nodes is finite due to the sporadic node density and node mobility. Moreover, the buffer space of each node is limited. Under this kind of resource constraints, i.e., the limited bandwidth and buffer space, it is a challenging task to

[4.1] The TTL used in this chapter is measured in the scale of time unit but not hop count.

deliver messages. The principle in designing the routing protocol is to maximize the message delivery probability, that is, to maximize the delivery probability of the message transfer within the limited contact duration and minimize the reduced delivery probability of the message dropping in buffer management.

This section first proposes the cooperative message transfer scheme, which considers the constraint of bandwidth. After that, the constraint of buffer space is further considered and the buffer management strategy is proposed. Finally, an enhancing strategy utilizing the extra bandwidth is proposed to improve the performance.

4.3.1 Cooperative Message Transfer Scheme

4.3.1.1 Broadcast and Phantom Message

In traditional DTN routing, the unicast is always adopted to exchange messages, in which the transmitted message can only be received by the specified receiver. However, the unicast cannot efficiently make use of the wireless channel. In contrast, the broadcast can fully utilize the natural property of the wireless channel to duplicate the message to all the neighbors. When a node transmits a message to one of its neighbors, it can add the receiver ID into the message and then broadcasts this message. After that, the specified receiver together with other neighbors can receive this message. Thus the broadcast can transfer the message to more receivers without extra bandwidth or energy consumption. Take the scenario in Figure 4.2(a) for example, when node A meets node E, it decides to transmit a message m_k to node E. By using unicast, only node E can receive m_k. However, if broadcast[4.2] is adopted, both nodes B and C can also receive m_k. As m_k is duplicated to more nodes, its delivery probability may increase. One underlying problem caused by the broadcast is that the duplication of the broadcasted message will increase its number of replicas, which may bring extra overhead into the network. To

[4.2] The broadcast here indicates one-hop message transfer, which is different from the concept of epidemic routing.

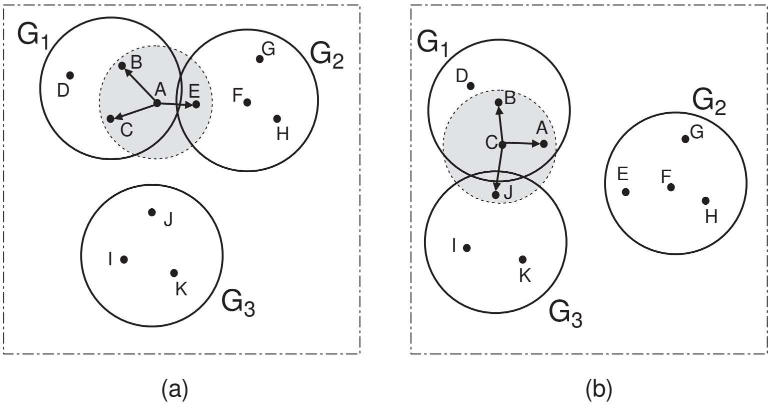


Figure 4.2. DTN routing using the broadcast and phantom message. The shadow region in (a) denotes the broadcast area of node A and the shadow region in (b) denotes the broadcast area of node C.

solve this problem, the *phantom message* is employed into the network. The phantom message refers to the message that will not be transferred between two encountering nodes. Only when the node with a phantom message meets the destination, then the phantom message can be transferred.

Each message m_k has a property attribute *prop* which will be set to 0 (or 1) if m_k is (or is not) a phantom message.

The *prop* attribute of each message is initiated as 1. When two encountering nodes exchange message, all other neighboring nodes overhear the broadcasted message and set their *prop* to 0, i.e., they will treat the overheard message as a phantom message. As the phantom message will not be transferred to other nodes except the destination, it will not bring extra overhead.

To increase the delivery probability, a node can deliver its phantom message to the destination when it encounters the destination. The generation of phantom message does not consume extra bandwidth or energy consumption, but it may still affect the storage of other messages when the buffer is full. To avoid this kind of negative impact, when a node with a full buffer needs to receive messages from its encounter, it will firstly delete its phantom message, the detail of which will be introduced in the buffer management strategy.

For example, in Figure 4.2(a), node A holds a message m_k with destination node J. When node A meets node E, it decides to disseminate some replicas of m_k to node E using broadcast. As the specified receiver, node E can receive m_k as a non-phantom message. Meanwhile, nodes B and C can also receive the broadcasted m_k as a phantom message since they are in the broadcast area of node A. As shown in Figure 4.2(b), when node C meets node J, as node C holds a phantom message of m_k, which is destined to node J, it can deliver m_k to node J directly. Thus, the direct delivery of the phantom message of m_k immediately makes m_k successfully delivered.

Though the phantom message does not bring extra bandwidth or energy consumption into the network, it may still occupy the node's storage which will affect the storage of other messages when the buffer space is used out. To avoid this negative effect, a node can delete its phantom message to free some buffer space when it needs to receive messages from its encounter while its buffer is fully used. The details will be introduced in the buffer management strategy subsection.

4.3.1.2 Estimation of Delivery Probability

This chapter will adopt the quota-based routing scheme, in which each message is initiated with a predefined number of replicas in the network. The initial number of replicas of each generated message is denoted as λ. Thus, a message will be considered as being successfully delivered if only one of the λ replicas arrives at the destination within the time-to-live (TTL).

In DTN routing, the prediction of the future contact information between each pair of nodes can be used when making the routing decision. As mentioned in the system model, there are n nodes in the network, in which each node can record the past meeting intervals between itself and other nodes. The set of recorded past meeting intervals between each pair of nodes v_i and v_j is denoted as $R_{ij} = \{\Delta t_1^{ij}, \Delta t_2^{ij}, \ldots, \Delta t_{r_{ij}}^{ij}\}$, where Δt_s^{ij} represents the past s^{th} meeting interval between nodes v_i and v_j ($\Delta t_s^{ij} = t_l^{ij}(s) - t_m^{ij}(s)$, where $t_l^{ij}(s)$ and $t_m^{ij}(s)$ are the s^{th} leaving time and meeting time between nodes v_i and v_j respectively, which are described in the system model) and r_{ij} is the total number of recorded meeting intervals between v_i and v_j. When v_i and v_j encounter, they will update the recorded past meeting interval between themselves. After that, node v_i can predict the meeting interval Δt^{ij} between itself and v_j as follows:

$$\Delta t^{ij} = \frac{1}{r_{ij}} \sum_{s=1}^{r_{ij}} \Delta t_s^{ij}.$$

Due to the exponential distribution [2][78], the probability $P_{ij}(T)$ that node v_i will encounter v_j within time interval T can be estimated as[4.3]:

$$P_{ij}(T) = 1 - exp(-\frac{T}{\Delta t^{ij}}) \tag{4.3.1}$$

As the geographical relationship among the nodes within the same group is close, their connections are relatively frequent. Thus, as long as the message is delivered into the destination group, it can almost guarantee that the message will be successfully delivered to the destination. Thus, for a message m_k held by node v_i with the time-to-live TTL_k, assume that the destination group is G_d, then node v_i can estimate the probability $P_i(m_k)$ that it can successfully deliver m_k as:

$$\begin{aligned} P_i(m_k) &= \max_{\forall v_j \in G_d} \{P_{ij}(TTL_k)\} \\ &= \max_{\forall v_j \in G_d} \{1 - exp(-\frac{TTL_k}{\Delta t^{ij}})\}. \end{aligned} \tag{4.3.2}$$

[4.3] Note that the calculation of the meeting probability $P_{ij}(T)$ itself does not rely on the assumption of exponential distribution of the inter meeting intervals.

4.3.1.3 Priority Order of Message Transfer

As the nodes in DTNs are always mobile, the inter-contact duration between each pair of encountering nodes is limited, which may be not long enough for them to exchange all the messages they hold. Thus, it is necessary for two encountering nodes to cooperate with each other to schedule the message transfer order, which can efficiently make use of the limited contact duration. When two nodes encounter, they will exchange the messages with the following priority order.

- First, all the messages (including the phantom messages) destined to the current encounter are transferred.
- Second, if the two nodes are from different groups, then all the messages (including the phantom messages) destined to the current encounter's group members are transferred. Otherwise, if the two nodes are from the same group, then all messages destined to other nodes within the same group are transferred.
- Third, for other messages with multiple replicas, the two encountering nodes will disseminate them according to the proportion of the estimated delivery probabilities. Note that no operation will be conducted for the messages held by both the two encountering nodes.
- Fourth, for the messages with single replica, each of the two encountering nodes will make the forwarding decision based on the estimated delivery probabilities.

When two nodes meet, the transfer of the message destined to the current encounter can immediately make this message be successfully delivered, so it has the highest priority to be transferred. As the connections among the nodes within the same group are relatively frequent, the messages destined to the current encounter's group members will have the second highest priorities if the two encounters are from different groups since the transfer of these messages can almost guarantee that they will be successfully delivered. Otherwise, if the two encountering nodes are within the same group, the messages destined to other nodes within this group will have the second highest priorities to be transferred, which can route them inside the group to the destinations.

After that, the messages with multiple replicas will be disseminated between two encountering nodes. Note that in this chapter, the dissemination indicates the distribution of replicas of a message between two encountering nodes, after which both the two nodes have the copy of the message, while the forwarding indicates the transfer of a message with single replica between two encountering nodes, after which the sender will not keep the copy of the message. Assume that nodes v_i and v_j hold a set of messages $\mathbb{M} = \{m_1, m_2, \ldots, m_N\}$[4.4] with multiple replicas and M_k ($M_k > 1$) represents the number of replicas of message m_k. The purpose of the dissemination procedure for the message with multiple replicas is to distribute these replicas to different nodes as soon as possible, which can increase the delivery probability. Thus, when nodes v_i and v_j encounter, they can sort the set of messages $\{m_1, m_2, \ldots, m_N\}$ with a descending order of the current number of replicas of each message, and the message with larger number of replicas has a higher priority to be disseminated. For the messages with the same number of replicas, their disseminating priorities will be determined based on the improved delivery probabilities of the dissemination. For example, assume that a set of messages $\mathbb{M}' \subset \mathbb{M}$ are with the same number of replicas, each of which is held by node v_i or v_j. For each message $m_k \in \mathbb{M}'$, the improved delivery probability $\Delta P(m_k)$ of the dissemination is:

$$\Delta P(m_k) = P_D(m_k) - P_{\overline{D}}(m_k) \tag{4.3.3}$$

[4.4] For each message of them, only node v_i or v_j holds it.

where $P_D(m_k)$ presents the delivery probability of m_k by v_i and v_j after the dissemination and $P_{\overline{D}}(m_k)$ presents the delivery probability of m_k by v_i and v_j before the dissemination. Here, $P_D(m_k) = 1 - (1 - P_i(m_k))(1 - P_j(m_k))$ and $P_{\overline{D}}(m_k) = a_i(m_k)P_i(m_k) + a_j(m_k)P_j(m_k)$, where $a_i(m_k)$ (or $a_j(m_k)$) is set to 1 if m_k is held by v_i (or v_j) before the dissemination; otherwise, $a_i(m_k)$ (or $a_j(m_k)$) is set to 0. Note that each message m_k is held by only one node (v_i or v_j), $a_i(m_k) + a_j(m_k) = 1$. Thus,

$$\Delta P(m_k) = a_j(m_k)P_i(m_k) + a_i(m_k)P_j(m_k) - P_i(m_k)P_j(m_k). \tag{4.3.4}$$

Thus, for the messages in $\mathbb{M}'$, their disseminating priorities can be determined based on the improved delivery probabilities of the dissemination and the message with the maximum improved delivery probability will be disseminated firstly.

To distribute the replicas of each message to different nodes as soon as possible, the messages with multiple replicas can be disseminated between two encountering nodes according to the proportion of their estimated delivery probabilities. For instance, when nodes v_i and v_j from different groups encounter, if node v_i holds a message m_k with M_k ($M_k > 1$) replicas and node v_j has no replica of m_k, then both nodes v_i and v_j can calculate their estimated delivery probabilities $P_i(m_k)$ and $P_j(m_k)$ for m_k using Eq. (4.3.2). After exchanging the estimated delivery probabilities with each other, v_i will determine to pass

$$\lfloor M_k \cdot \frac{P_j(m_k)}{P_i(m_k) + P_j(m_k)} \rfloor \tag{4.3.5}$$

replicas of message m_k to v_j.

The messages with single replica have the lowest priorities. The method to determine the forwarding priorities of the messages with single replica is similar with that of the multiple replicas case, which is based on the improved delivery probabilities of the forwarding. For instance, if nodes v_i and v_j hold a set of messages $\mathbb{M}''$ with single replica (each message in $\mathbb{M}''$

is held by only one of the two nodes), then for each message $m_k \in \mathbb{M}''$, the improved delivery probability $\Delta P(m_k)$ of the forwarding is:

$$\Delta P(m_k) = P_F(m_k) - P_{\overline{F}}(m_k), \tag{4.3.6}$$

where $P_F(m_k)$ presents the delivery probability of m_k by v_i and v_j after the forwarding and $P_{\overline{F}}(m_k)$ presents the delivery probability of m_k by v_i and v_j before the forwarding. As $P_F(m_k) = a_j(m_k)P_i(m_k) + a_i(m_k)P_j(m_k)$ and $P_{\overline{F}}(m_k) = a_i(m_k)P_i(m_k) + a_j(m_k)P_j(m_k)$, we can get

$$\Delta P(m_k) = (a_i(m_k) - a_j(m_k))(P_j(m_k) - P_i(m_k)), \tag{4.3.7}$$

where $a_i(m_k)$ (or $a_j(m_k)$) is set to 1 if m_k is held by v_i (or v_j) before the forwarding; otherwise, $a_i(m_k)$ (or $a_j(m_k)$) is set to 0.

Thus, for the messages in $\mathbb{M}''$, their forwarding priorities can be determined based on the improved delivery probabilities of the forwarding, which can be calculated using Eq. (4.3.7).

After calculating the forwarding priorities of the messages with single replica, the node will decide whether to forward each of them to the encounter or not. As the forwarding of the message with single replica will consume the limited bandwidth, a threshold δ_P is employed in the forwarding decision to balance the tradeoff between the overhead and delivery ratio. For example, when nodes v_i and v_j encounter, and a single-replica message m_k is with the highest forwarding priority held by node v_i, then both of the two nodes will calculate their estimated delivery probabilities $P_i(m_k)$ and $P_j(m_k)$ for m_k using Eq. (4.3.2). If:

$$\Delta P(m_k) = P_j(m_k) - P_i(m_k) > \delta_P, \tag{4.3.8}$$

then node v_i will forward m_k to v_j. Otherwise, there will be no operation for m_k between the two nodes.

4.3.1.4 Cooperative Message Transfer Algorithm

In this section, a transfer priority function is defined to calculate the transfer priorities of all the messages to be exchanged between two encountering nodes. When nodes v_i and v_j encounter, the transfer priority function of message m_k, denoted as $f_{TP}(m_k)$, is defined as follows:

$$f_{TP}(m_k) = \begin{cases} \lambda + 2, & \text{if } m_k \text{ is destined to the encounter;} \\ \lambda + 1, & \text{if } m_k \text{ is destined to the encounter's} \\ & \quad \text{group member;} \\ M_k + \Delta P(m_k), & \text{elsewhere.} \end{cases} \tag{4.3.9}$$

where λ is the number of replicas of m_k when generated and M_k is the current residual number of replicas of m_k. If $M_k > 1$, $\Delta P(m_k)$ can be calculated using Eq. (4.3.4). Otherwise, if $M_k = 1$, $\Delta P(m_k)$ can be calculated using Eq. (4.3.7). It is obvious that $M_k \leq \lambda$ and $0 \leq \Delta P(m_k) \leq 1$. Thus, if m_k is destined to the encounter, its transfer priority will be set as $\lambda + 2$, which indicates that it has the highest priority to be transferred. And if m_k is destined to the encounter's group member, its transfer priority function will be set as $\lambda + 1$, which indicates that it has the second highest priority to be transferred. Otherwise, the transfer priority of m_k will be set as $M_k + \Delta P(m_k)$. Since $M_k \leq \lambda$ and $\Delta P(m_k) \leq 0$, it is obvious that $M_k + \Delta P(m_k) \leq \lambda + 1 < \lambda + 2$, which indicates that m_k will have the lowest transfer priority ($M_k > 1$ for the multiple replicas case and $M_k = 1$ for the single replica case).

Note that only one of the two encounters needs to schedule the message transfer sequence, which is conducted by the node with a smaller ID in the implementation. When nodes v_i and v_j encounter, the procedure of the cooperative message transfer between them is described in Algorithm 5.

Example. Figure 4.3 illustrates the cooperative message transfer procedure between two encountering nodes S and A for the sample DTN given in Figure 4.1(b). Assume that the initial number of replicas is 10, i.e., $\lambda = 10$. Before the two nodes meet, node S holds messages m_1, m_2, m_3 and m_4. The destination of m_1 is node D and node S has 3 replicas of m_1. The

Algorithm 5 Cooperative Message Transfer Algorithm

```
Nodes $v_i$ and $v_j$ update their contact histories and calculate the up-to-date average meeting interval.
Nodes $v_i$ and $v_j$ calculate the transfer priorities of their held messages and exchange the priorities with each other.
Nodes $v_i$ and $v_j$ exchange the messages with a descending order of transfer priorities of their held messages.
```

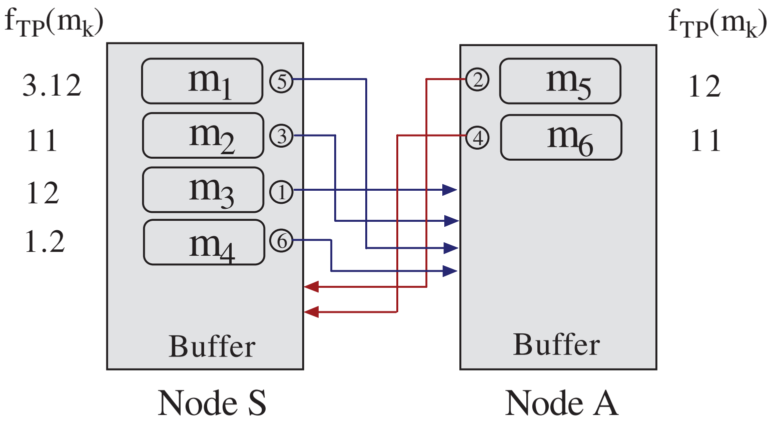


Figure 4.3. The cooperative message transfer procedure between two encountering nodes.

destinations of m_2 and m_3 are nodes B and A respectively. And node S only has one replica of m_4 with the destination node C. The other node A holds messages m_5 and m_6 with the destinations nodes S and E respectively. When nodes S and A encounter, they can calculate the transfer priorities as follows: $f_{TP}(m_1) = M_1 + \Delta P(m_1) = 3 + \Delta P(m_1)$, $f_{TP}(m_2) = \lambda + 1 = 11$, $f_{TP}(m_3) = \lambda + 2 = 12$, $f_{TP}(m_4) = M_4 + \Delta P(m_4) = 1 + \Delta P(m_4)$, $f_{TP}(m_5) = \lambda + 2 = 12$ and $f_{TP}(m_6) = \lambda + 1 = 11$. Note that $0 \leq \Delta P(m_1), \Delta P(m_4) \leq 1$. Thus, nodes S and A will firstly exchange m_3 and m_5 and then exchange m_2 and m_6. After that, they will exchange m_1. Assume the estimated probability that node S can successfully deliver m_1 is $P_S(m_1) = 0.3$ and the estimated probability that node A can successfully deliver m_1 is $P_A(m_1) = 0.6$, thus $\Delta P(m_1) = 1-(1-P_A(m_1))(1-P_S(m_1))-P_A(m_1) = 1-(1-0.6)(1-0.3)-0.6 = 0.12$ and $f_{TP}(m_1) = 3.12$, then node S will pass $\lfloor M_1 \cdot \frac{P_A(m_1)}{P_S(m_1)+P_A(m_1)} \rfloor = \lfloor 3 \cdot \frac{0.6}{0.3+0.6} \rfloor = 2$ replicas of m_1 to node A. Finally, assume $P_S(m_4) = 0.4$ and $P_A(m_4) = 0.6$, thus $\Delta P(m_4) = P_A(m_4) - P_S(m_4) = 0.6 - 0.4 = 0.2$ and $f_{TP}(m_4) = 1.2$. So m_4 is with the lowest transfer priority and node S will determine whether to forward m_4 to node A by comparing $\Delta P(m_4)$ with the threshold δ_P.

4.3.2 Buffer Management Strategy

In DTNs, the store-carry-and-forward mechanism is always adopted. Also, each message may have multiple replicas in the network. The combination of the two factors may make the buffer of each node full due to the limited buffer space. Thus, it is necessary to design an efficient buffer management strategy. The difference between utilizing the limited bandwidth and buffer space is that the messages that cannot be transferred in current transfer opportunity may get another transfer opportunity in next contact, while the message that is dropped from the buffer will have no transfer opportunity. Thus, the criteria of the buffer management strategy is to reduce the negative impacts of the message dropping on the delivery probability, that is, to avoid dropping the *important* messages.

In the previously proposed cooperative message transfer scheme, the phantom message is generated without extra bandwidth or energy consumption and the delivery probability can be increased by directly delivering the phantom message to the destination. However, under the constraint of buffer space, the storage of the phantom messages will consume some of the limited buffer space, their existence may affect the delivery of other messages. Thus, to avoid the negative impacts of the phantom message, when the buffer is full, each node will firstly drop the phantom message to vacate the buffer space to receive other messages from its encounter. As the phantom message is some kind of bonus of the broadcast, dropping the phantom message will not severely affect the delivery probability. However, if the node holds not any phantom message, then it can adopt the following cooperative caching scheme.

As the nodes within the same group have a close geographical relationship and their connections are relatively frequent, they can determine to cooperatively cache the messages. Since the transfer of message will bring extra overhead, i.e., the bandwidth and energy consumption, a cooperative message caching scheme using the phantom message, which can be conducted without really transferring the message, is proposed. When the buffer of a node is full, it can check with its neighbors whether they hold a phantom message whose corresponding non-phantom message is held by themselves. For example, as shown in Figure 4.2(a), when node A with a full buffer needs to receive messages from node E and all the messages held by node A are non-phantom messages, node A can enquire its neighbors, i.e., nodes B, C and E whether they hold a phantom message whose corresponding non-phantom message is held by node A. For instance, suppose node A has message m_k, and node B has a phantom message of m_k. Then node A can send a control message to inform node B to transform the phantom message of m_k into a non-phantom one (i.e., change the property of the phantom message from 0 to 1) with the same number of replicas of the non-phantom message m_k in node A. After that, node A can delete m_k to vacate buffer space to receive other message from node E. Thus, it can achieve the cooperative message caching between nodes A and E without transferring a real message. Therefore, it dose not require extra bandwidth or energy consumption[4.5] and the dropped message is actually a phantom one, which will not severely impact the delivery probability.

If the node still does not have enough buffer space to receive message from its encounter, it can achieve the cooperative message caching by forwarding one of the messages to any of its neighboring nodes with spare buffer space. To select the message (with multiple replicas or single replica) to be forwarded, the node can refer to the improved delivery probability of forwarding, which can be calculated using Eq. (4.3.7). Thus, the node will forward the message with the largest improved delivery probability to its neighbor to vacate the buffer space.

However, if none of its neighbors has spare buffer space, the node has to directly drop one of the messages from the buffer. To reduce the negative impact of message dropping on the

[4.5] As the control message is very short compared to that of the message, the overhead brought by the control message can be ignored.

delivery probability, an occupying priority function for each message is defined to determine their priorities to stay in the buffer. The occupying priority function f_{OP} of each message m_k in node v_i is defined as follows:

$$f_{OP}(m_k) = M_k + P_i(m_k) \tag{4.3.10}$$

where M_k is the current residual number of replicas of m_k in node v_i and $P_i(m_k)$ is the estimated probability that node v_i can successfully deliver m_k, which is formulated in Eq. (4.3.2). Obviously, $M_k \leq \lambda$ and $0 \leq P_i(m_k) \leq 1$. Thus, according to Eq. (4.3.10), the message with a larger number of replicas has a higher occupying priority. And for the messages with the same number of replicas, the one with a higher estimated delivery probability has a higher occupying priority. After calculating the occupying priorities of all the messages, the node will firstly drop the one with the lowest occupying priority.

4.3.3 Enhancing Strategy With Extra Bandwidth

The phantom messages are the extra information obtained due to the natural property of broadcast and the direct delivery of the phantom messages to the destinations can increase their delivery probabilities. However, if two encountering nodes are still connected after exchanging all the messages, they can further utilize this contact duration, that is, transform the held phantom messages into non-phantom messages and disseminate these non-phantom messages between them.

The procedure of transforming the phantom message into non-phantom message is defined as the activation of phantom message. Before the activation of phantom message, one of its corresponding non-phantom messages in the network has to be transformed into phantom message. Furthermore, the number of replicas of the transformed message (the previous phantom message before the transformation) will be set to the number of replicas of the transformed phantom message (the previous non-phantom message before the transformation). However, due to the distributed property of DTNs, the activation of phantom message can be conducted only if one of the node's neighbors has a corresponding non-phantom message. The node can conduct the activation of phantom message after sending a control message to inform the neighbor to transform the corresponding non-phantom message into a phantom one. Therefore, the number of replicas of the corresponding non-phantom message will not change after the activation.

For example, as shown in Figure 4.2(a), node A has a message m_k with M_k replicas and the destination is not in group G_1, G_2 or G_3. When node A meets node E, it will disseminate some replicas of m_k to node E, which is based on the proportion of their estimated delivery probabilities. As node A uses the broadcast to disseminate m_k to node E, both nodes B and C can overhear m_k as a phantom message. After that, when node C meets node J as shown in Figure 4.2(b), it will first exchange the messages with node J. If nodes C and J are still connected after exchanging all the messages, node C can further transform the phantom message m_k in its buffer into a non-phantom one, before which it has to send a control message to inform node A to transform message m_k into a phantom message. Also, the number of replicas of the transformed non-phantom message m_k in node C should be set to that of replicas of the original message m_k in node A. After the activation of phantom message m_k, node C can efficiently utilize the extra contact duration by disseminating m_k with node J based on the proportion of their estimated delivery probabilities for m_k.

4.4 Performance Evaluation

In this section, the proposed group aware cooperative routing protocol will be evaluated with the three performance metrics: message delivery ratio, latency and goodput in the Opportunistic Network Environment simulator (ONE) [33]. The effectiveness of the proposed routing protocol will be illustrated by comparing it with other existing DTN routing protocols. The effects of the network parameters on the routing performance of the proposed routing protocol will also be analyzed.

Table 4.1. Simulation settings.

Parameter name	Default	Range
Number of nodes in the network (n)		60~160
Number of nodes in each group		4~8
Initial number of replicas of each message (λ)	10	6~12
Threshold used in the forwarding (δ_P)	0.5	
Time to live of each message (TTL)	20 mins	
Size of each message (S_m)	100 KB	
Buffer size of each node (S_b)	1 MB	
Transmission rate (T_r)	250 KBps	
Transmission range (R)	100 m	
Message generation rate	2 per min	

4.4.1 Performance Metrics and Simulation Settings

The three performance metrics will be employed into the performance evaluation, including the delivery ratio, latency and goodput. The definitions of these three metrics are shown as follows:

- *Delivery ratio:* The ratio of the number of the successfully delivered messages to the number of all the generated messages.
- *Latency:* The average end-to-end delivery delay between each pair of source and destination in the network.
- *Goodput:* The ratio of the number of the successfully delivered messages to the number of relayed messages in the network.

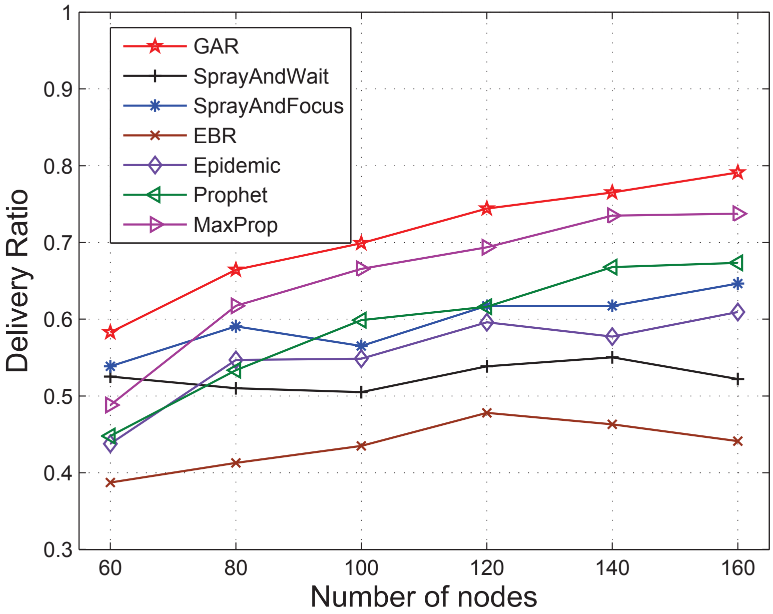


Figure 4.4. Delivery ratio comparison between the proposed GAR protocol and other existing protocols.

In the simulations, the Reference Point Group Mobility (RPGM) [68] is employed as the mobility model. Each group has a logical "center", the motion of which defines the entire group's motion behavior. And the movement of each group center follows the random waypoint mobility model. The moving process of each node is the combination of its group movement with a random motion vector, which allows the independent random motion behavior of each node within the group. The movement of each group is independent, and the random motion vector for a node inside the group is also independent with that of other nodes in the same group. In the implementation of simulations, the logical center of each group moves with the speed varying from 2.7 to 13.9 m/s within an area of $4500m \times 3400m$. Each simulation lasts for $10000s$. The other simulation settings are listed in Table 4.1.

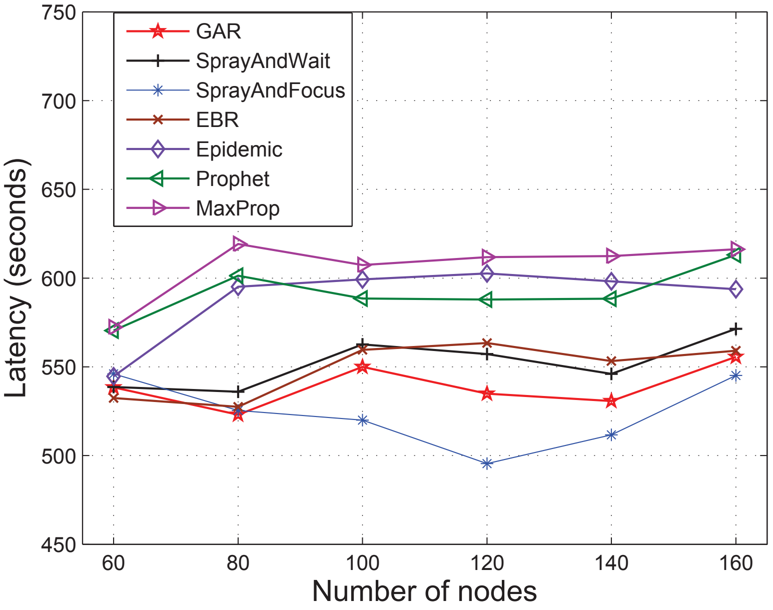


Figure 4.5. Latency comparison between the proposed GAR protocol and other existing protocols.

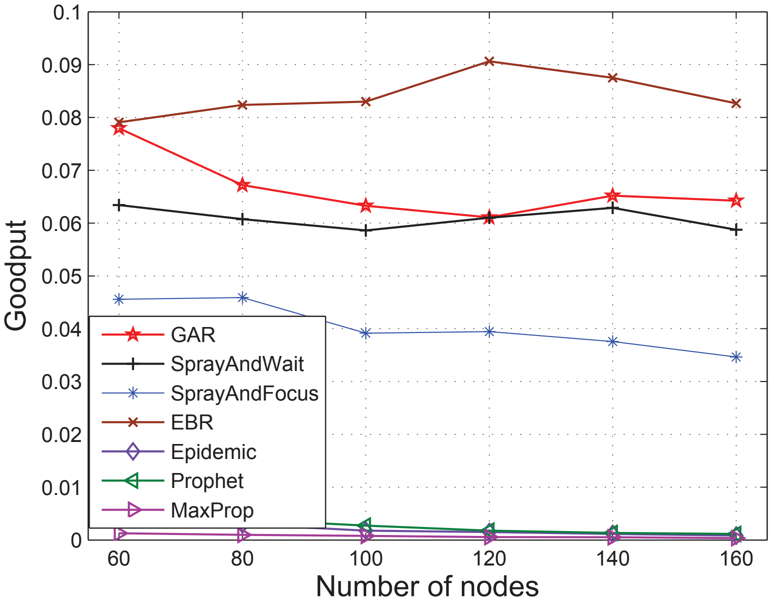


Figure 4.6. Goodput comparison between the proposed GAR protocol and other existing protocols.

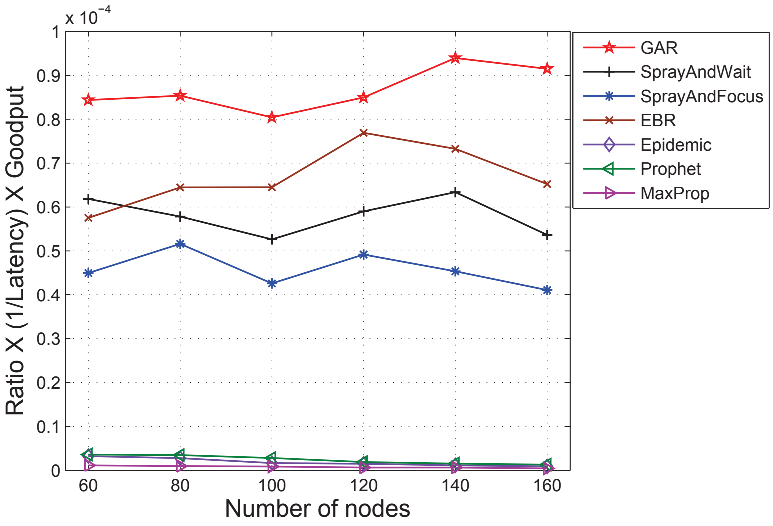


Figure 4.7. Composite performance comparison between the proposed GAR protocol and other existing protocols.

4.4.2 Simulation Results

To evaluate the effectiveness of the proposed group aware cooperative routing protocol — GAR, this section will compare it with other six popular protocols: Epidemic [84], Prophet [46], MaxProp [5], Spray-and-Wait [78], Spray-and-Focus [79] and EBR [57]. After that the effects of λ, the initial number of replicas of each message, on the performance of the proposed GAR protocol will be analyzed.

Figure 4.4, Figure 4.5 and Figure 4.6 show the performance comparison between the GAR and other six protocols. λ is set as 10 for the GAR, Spray-and-Wait, Spray-and-Focus and EBR protocols. Figure 4.4 shows the delivery ratio of the seven protocols. The figure shows that the GAR protocol achieves the highest delivery ratio, even higher than those of the epidemic based protocols. As in the DTN with resource constraints, only a small set of messages can be transferred during a single contact and the buffer space is easy to be full, the epidemic based protocols have not carefully considered these problems. Thus, their delivery ratios are lower than that of GAR. The delivery ratio of the EBR protocol is the lowest. Figure 4.5 shows the latency of the protocols. The GAR protocol achieves the second shortest latency. And the

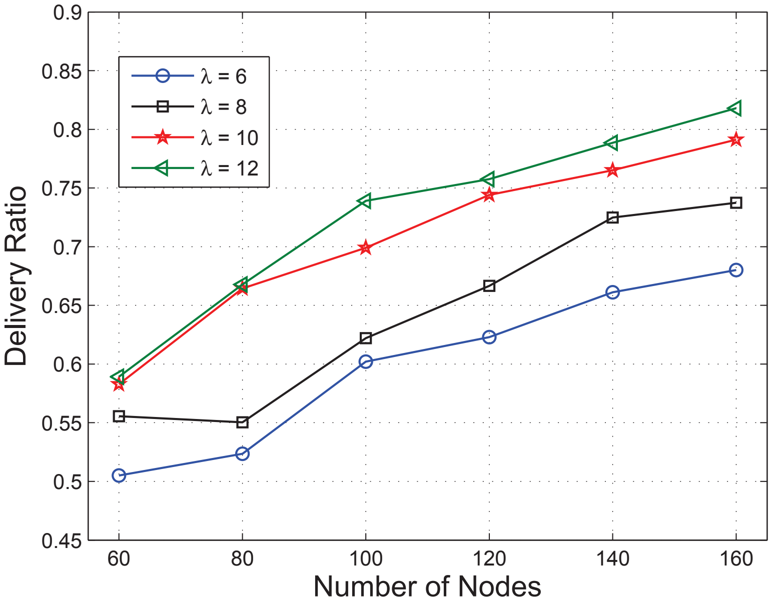


Figure 4.8. Effects of the value of λ on delivery ratio of the proposed GAR protocol.

latency of epidemic based protocols is longer than that of the quota based protocols. Figure 4.6 shows the goodput of the protocols. Obviously, the epidemic based protocols require the largest number of transfer so their goodput is lower than those of other protocols. The goodput of the EBR protocol is the highest since the nodes in the EBR are seldom to transfer the messages (thus the delivery ratio of EBR is the lowest). The GAR protocol achieves the second highest goodput. Overall, the proposed GAR protocol performs effectively compared with other six protocols.

Figure 4.7 shows the composite performance comparison between the GAR protocol and other six protocols. A composite metric is proposed in [57] to evaluate the overall performance of a specific protocol. The composite metric is defined as $Ratio \times (1/Latency) \times Goodput$, which encourages the one with a high message delivery ratio and high goodput and penalizes it with a long end-to-end delivery delay. The figure illustrates that the proposed GAR protocol outperforms other protocols.

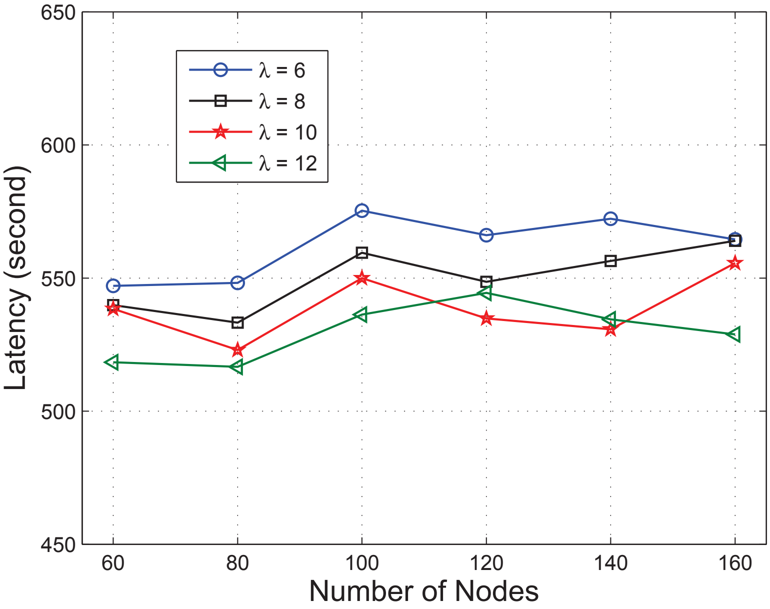


Figure 4.9. Effects of the value of λ on latency of the proposed GAR protocol.

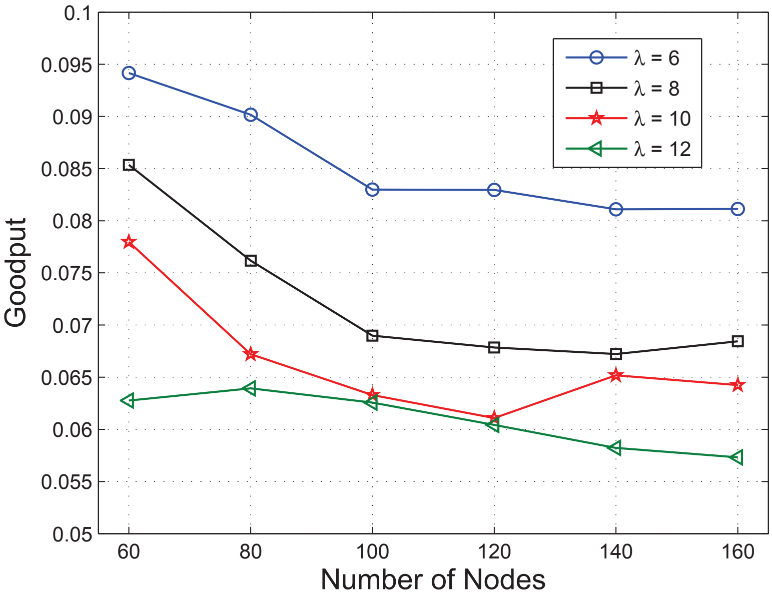


Figure 4.10. Effects of the value of λ on goodput of the proposed GAR protocol.

Figure 4.8, Figure 4.9 and Figure 4.10 show the effects of λ on the performance of the proposed GAR protocol. The value of λ varies from 6 to 12 with an increment of 2. The delivery ratio of the GAR protocol rises when the value of λ increases because each message is considered to be successfully delivered if only one of its replicas arrives at the destination within the TTL and the increase of λ can increase the delivery probability of each message. The increase of λ will reduce the latency. However, the increase of λ can heighten the overhead as it will cause a larger amount of transfers in the network thus the GAR protocol will achieve a lower goodput. Therefore, it is a tradeoff to determine an appropriate value of λ.

4.5 Summary

This chapter first introduces the system model for the delay tolerant networks with group feature. By applying the group feature in the network, a cooperative routing protocol including the cooperative message transfer scheme and buffer management strategy is proposed. In the cooperative message transfer scheme, the constraint of bandwidth is considered and the message transfer priorities are designed to maximize the delivery probability. In the buffer management strategy, by considering the constraint of buffer space, the cooperative message caching scheme is proposed and the dropping order of the messages is designed to minimize the reduced delivery probability. Finally, the proposed group aware cooperative routing protocol is implemented in the ONE simulator and the simulations are conducted to illustrate its effectiveness under different network parameters.

CHAPTER 5

SECURE COOPERATION INCENTIVE MECHANISMS FOR MESSAGE FORWARDING IN NON-COOPERATIVE DTNS

5.1 Introduction

As an emergent communication paradigm, delay tolerant networks (DTNs) can be competent for many applications such as social networks[45, 26], vehicular networks [5, 42, 10], pocket switched networks [73, 7], and habitat monitoring sensor networks [55, 90], etc. The nodes in DTNs are intermittently connected due to the high mobility and sparse deployment of the nodes, which makes the traditional routing protocols not applicable. Epidemic routing [84] is a simple but competitive DTN routing protocol, in which each node will make use of every contact opportunity to replicate the messages to its encounter. Although the epidemic routing may introduce a high overhead, it can be easily implemented in applications and achieve satisfying performance [42, 8].

However, the nodes in DTNs may be managed by some rational individuals, such as the human-beings or other autonomous parties. Such nodes may be selfish [42, 8, 94, 10], i.e., they may decline to cooperate with others to truthfully forward the messages if they cannot make any profit from the message forwarding; or they may even conduct some malicious behaviors to get extra profit from the message forwarding. Although Some incentive mechanisms [25, 94, 41], which requires the path from the source to destination to be stable for most of the time, have been proposed to motivate the cooperation for the nodes in the wireless ad hoc networks, this kind of mechanisms cannot be applicable in the DTNs since the nodes are intermittently connected and the path from the source to destination is not always existing. Due to the distributed characteristic of DTNs, it is a challenging issue to detect and prohibit the selfish behaviors conducted by the individual nodes. Therefore, it is necessary to have efficient cooperation incentive mechanisms for the non-cooperative DTNs to motivate the selfish nodes to cooperate with each other in the message forwarding. The non-cooperative DTN is defined as a kind of DTNs in which some of the nodes are selfish and will not forward the messages for other nodes if they cannot get any benefit.

Generally, the incentive mechanisms can be classified into two categories [83]: reputation-based schemes and credit-based schemes. The reputation-based schemes require each node to monitor the traffic information of all its neighbors and keep track of their reputation values, which should be propagated to all other nodes efficiently and effectively. This is quite challenging for DTNs due to their intermittent connectivity. On the other hand, credit-based schemes use virtual credits to motivate selfish nodes to participate in the message forwarding; the credits they earned from forwarding other nodes' messages can be used to pay for the delivery of their own ones. It is obvious that the credit-based schemes better adapt to the intermittent connective characteristic of the DTNs.

MobiCent [8] is a recently proposed credit-based incentive scheme for DTN routing, in which the payment mechanism is based on the hop count of the delivery path. MobiCent is incentive compatible (i.e., the truthful cooperation is adopted by each of the nodes) and it can guarantee that the payment for each delivered message is upper bounded, which is a typical drawback of the VCG-based (Vickrey-Clarke-Groves) incentive schemes [94, 41]. It also defends against the security threat of the *edge insertion attack* (a kind of sybil attack), which cannot be solved in SMART [95]. However, MobiCent suffers the *edge removal attack*, in which a node can get extra payoff from the system by removing the records of other nodes' behaviors of relaying. This motivates us to design secure cooperation incentive mechanisms to simultaneously resolve the above drawbacks of the previous credit-based incentive schemes.

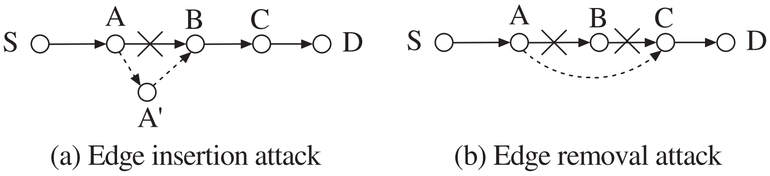
In this chapter, the following three typical malicious behaviors of the selfish nodes are considered, which can seriously disturb the incentive schemes.

1. *Edge insertion attack (or sybil attack):* A selfish node may attempt to forge a virtual edge (or a sybil node) into a path to get extra reward from the system. As shown in Figure 5.1(a), node A may forge a sybil node A' between node B and itself, and then the reward to node A' will be essentially obtained by node A.

2. *Edge removal attack:* A selfish node may attempt to remove or hide the behaviors of relaying of other nodes in a path to get more reward, which should be originally paid to the removed nodes. As shown in Figure 5.1(b), node C may remove the edges $\overrightarrow{AB}$ and $\overrightarrow{BC}$ and cheat the system that the message it has received is directly from node A, but not node B, then node B will not get the reward, which benefits node C.

3. *Content modification attack:* A selfish node, or several colluding nodes, may attempt to modify the content of the report message, which contains the path information such as the transmitting time and receiving time. As the reward is calculated based on the information of the report message, modifying the content may alter the reward for each node.

The design goals of the proposed cooperation incentive mechanisms in this chapter are threefold: 1) incentive compatibility: to make the truthful forwarding be the dominant strategy



(a) Edge insertion attack (b) Edge removal attack

Figure 5.1. Illustrations of the edge insertion attack and edge removal attack.

for all the nodes; 2) budget control: to guarantee that the payment for each delivered message is upper bounded; 3) security enhancement: to defend against the above typical attacks in the cooperation incentive mechanisms. By considering the design goals, a credit-based rewarding scheme called earliest path singular rewarding (EPSR) scheme is first proposed to motivate the nodes to truthfully forward the messages during every contact opportunity. By further considering that a node may get more contact information of others and misbehave accordingly to take advantage of that, another credit-based rewarding scheme called earliest path cumulative rewarding (EPCR) scheme is then proposed. The main idea of the proposed rewarding schemes is to reward each node in the earliest delivery path according to its *contribution time*, which is the period of time that the node holds the message.

5.2 System Model

This chapter considers a DTN formed by a set of mobile nodes. The connectivity among the network is intermittent due to the mobility and sparse deployment of the nodes. When two nodes encounter each other, they will exchange the messages in their buffers in the order according to the priority of each message. Various DTN routing protocols have different message priorities. For the epidemic routing protocol, all messages have the same priority. A node will make use of every contact opportunity to replicate its held messages to its encounter.

To facilitate the description of the proposed rewarding schemes, this section first gives the following definitions: An *edge* $e = (\{v_i, v_j\}, t(e))$ is used to present an opportunistic link between two encountering nodes, where $\{v_i, v_j\}$ are the two nodes and $t(e)$ is the contact time. It is denoted that $v_i \in e$ and $v_j \in e$ for the edge $e = (\{v_i, v_j\}, t(e))$. A *delivery path* $P = \{e_1, e_2, ..., e_m\}$ is used to present a message forwarding path from a source node to a destination node, where $e_1, e_2, ..., e_m$ are a sequence of edges listed in a non-decreasing order of the contact time. Two adjacent edges share a common node. The *delivery time* of a message in a path, $t(P)$, is defined as the contact time of the last edge in the path, i.e., $t(P) = \max_{e_i \in P} \; t(e_i)$. A node

$v_i \in P$ if there exists $e_j \in P$ such that $v_i \in e_j$. The corresponding notations in this paper are defined in TABLE 5.1.

Figure 5.2 shows a sample DTN. $e = (\{A, B\}, 10:35am)$ denotes a link between nodes A and B at 10:35am. The path $S \rightarrow A \rightarrow B \rightarrow C \rightarrow D$ is denoted as $P = \{(\{S, A\}, 10:15am), (\{A, B\}, 10:35am), (\{B, C\}, 10:40am), (\{C, D\}, 10:45am)\}$. The delivery time of a message from nodes S to D in this path is 10:45am. Nodes S,A,B,C,D are all $\in P$.

Table 5.1. Terminology.

Notation	Definition
e	A delivery edge
t(e)	The contact time of edge e
P	A delivery path
t(P)	The delivery time of path P
TTP	Trusted third party
VB	Virtual bank
TTL	Time-to-live of the messages
$Ct(v_i, P)$	Contribution time of node v_i in path P
PK_i	Public key of node v_i
RK_i	Private key of node v_i
$Rd(v_i, m_k)$	The reward to node v_i for the delivery of message m_k
t_i^t	Transmitting time of node v_i
t_i^r	Receiving time of node v_i
$Pay(S, m_k)$	Payment of source node S for the delivery of message m_k
$Fn(P_i, P_j)$	Furcation node of two paths P_i and P_j
T	The time period to transmit a unified-size message
$\mathbb{P}^k$	The set of all delivery paths of message m_k

As a totally distributed network, a DTN requires a central controller, called *Trusted Third Party* (TTP), to manage the rewarding process, i.e., to store the key information of the nodes and provide verification and payment services for the nodes that participate in the message forwarding. Each node can use the short-range, high-bandwidth links to exchange messages with its encounters; it can also use the long-range, low-bandwidth links to communicate with the TTP for the verification and payment during the rewarding process. A Virtual Bank (VB)

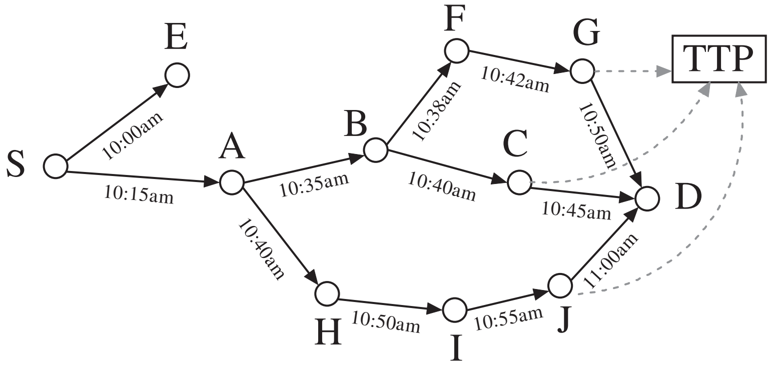


Figure 5.2. An example of the rewarding process for the message forwarding in a DTN. A message is generated by node S at 9:55am. The solid-line arrows and the dashed-line arrows indicate the short-range, high-bandwidth links and the long-range, low-bandwidth links respectively. The time-stamp under each solid-line arrow indicates the transmitting time of the message.

is also needed, which is managed by the TTP. the VB provides each node an account for saving its virtual credits earned from the message forwarding operations.

In the cooperation incentive mechanisms, each relay node can add some path information including its ID, receiver ID, receiving time and transmitting time onto the message when forwarding the message. After the message reaches the destination, the last intermediate node will extract the path information from the message and submit the path information as a *report message* to the TTP. The TTP calculates the rewarding credits for each relay node based on the report message and charges the source node for the corresponding credits. As shown in Figure 5.2, node S generates a message m_k for the destination node D. m_k will be delivered to node D by nodes C, G and J along three different paths respectively. Nodes C, G and J will also submit the report messages to the TTP. The TTP can then conduct the rewarding process to reward each node according to the specific rewarding scheme.

5.3 Earliest Path Singular Rewarding Scheme

This section first introduces the contribution metric to be used in the proposed rewarding schemes. Then the earliest path singular rewarding (EPSR) scheme is proposed. Finally, the analysis on the proposed EPSR scheme is given.

5.3.1 Contribution Metric

In some of the previous credit-based incentive schemes, the amount of reward given to each node is determined by the hop count of the delivery path. However, such a rewarding scheme can be easily attacked by the edge insertion attack or edge removal attack. Take MobiCent [8] for example, as shown in Figure 5.2, the node S generates a message with the destination node D. The message is delivered to the destination along three paths, among which path $P = S \rightarrow A \rightarrow B \rightarrow C \rightarrow D$ is with the least hop count, then each node in path P will get a reward of $(2+\epsilon)^{N-n}$ cents, where N is the maximum hop count of the path, n is the hop count of path P and ϵ is a small positive natural number. Suppose $N = 5$. Thus, $n = 4$. So each of nodes A, B, C and D will get a reward of $(2+\epsilon)^{N-n} = (2+\epsilon)$ cents. However, the selfish nodes can conduct the edge removal attack to get more credits from the MobiCent. For instance, nodes A and B can collude together and act as a single node (this is a kind of edge removal attack since the edge $A \rightarrow B$ is removed from the path), then the hop count of path P will be 3 and they will get a total reward of $(2+\epsilon)^2$ cents. Finally, each of nodes A and B will get half of the total reward, i.e., $(2+\epsilon)^2/2 = (1+\epsilon/2)(2+\epsilon)$ cents, which are larger than the original reward.

To motivate the nodes in the message forwarding, the credit-based incentive schemes will reward the nodes that contribute to the delivery of a message based on certain contribution metric. In the proposed scheme, the *contribution time* will be chosen as the contribution metric to measure the contribution of a node on the delivery of a message. The contribution time of a node in a delivery path of message m_k starts from the moment this node receives m_k and ends at the moment it transmits m_k to the next node in the path. It is easy to see that the contribution time of the source node begins from the moment the message is generated and the destination node has no contribution time in the path. For an intermediate node v_i in a source-to-destination path P, its contribution time in P can be denoted as $Ct(v_i, P)$, which can be formulated as:

$$Ct(v_i, P) = \max_{v_i \in e_j, e_j \in P} t(e_j) - \min_{v_i \in e_j, e_j \in P} t(e_j) \tag{5.3.1}$$

For the sample DTN shown in Figure 5.2, it is assumed that node S generates a message for destination node D at 9:55am. After the delivery of the message, the contribution time of nodes S, A, B and C in the path $S \rightarrow A \rightarrow B \rightarrow C \rightarrow D$ is 20 minutes (=10:15am−9:55am), 20 minutes (=10:35am−10:15am), 5 minutes (=10:40am−10:35am) and 5 minutes (=10:45am−10:40am), respectively.

Suppose that a message is generated at a source node with a time-to-live (TTL)[5.1]. The message has been delivered along a path P to the destination and the residual time of the message when it arrives at the destination is t_r. It can be easily obtained that:

$$\sum_{\forall v_i \in P} Ct(v_i, P) = TTL - t_r. \tag{5.3.2}$$

Since the nodes in a path cannot easily modify the total contribution time without modifying other nodes' contribution time, this section takes advantage of this property by using the contribution time of each node as the rewarding basis.

5.3.2 EPSR Scheme

As the nodes are motivated to forward the messages during every contact, the message generated by the source node will be delivered to the destination node via different paths. The

[5.1] The TTL used in this chapter is measured in the scale of time unit but not hop count.

main idea of the EPSR scheme is to reward the nodes only in the earliest delivery path of the message. The reward to each node is calculated based on its contribution time in the earliest delivery path of the message and all the rewards for the successful delivery of the message are paid by the source node.

The EPSR scheme has four phases: system initialization, message generation, message forwarding, and rewarding and charging.

5.3.2.1 System Initialization

As the rewarding procedure will be conducted by the TTP based on the submitted report messages, it is significant to protect the security of the report messages to prevent the contribution time from being modified (by the content modification attack). In this chapter, the asymmetric cryptographic method [34] will adopted to encrypt the report messages. In the initialization phase, the system generates several pairs of public/private keys. The TTP holds the private keys and distributes the public keys to the nodes. Thus, the TTP acquires the information of the corresponding relationships between the private keys and nodes. The public key of node v_i and the corresponding private key of v_i, which is obtained by the TTP, is denoted as PK_i and RK_i respectively. Note that it is not required that each node has an unique public key, that is, PK_i may be the same with PK_j where $i \neq j$. Thus, the system does not need a large number of key pairs, which is impractical for a large-scale network. The encrypted message m_k using the public key PK_i is also denoted as $Enc(PK_i, m_k)$ and the decrypted message m_k using the private key RK_i is denoted as $Dec(RK_i, m_k)$. Obviously, $Dec(RK_i, Enc(PK_i, m_k)) = m_k$.

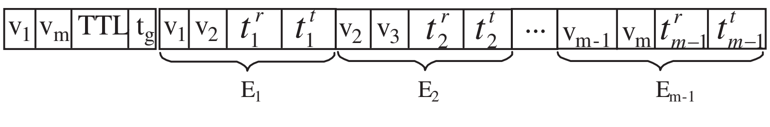


Figure 5.3. A contribution table of the delivery path $v_1 \rightarrow v_2 \rightarrow \cdots \rightarrow v_m$ with the source node v_1 and destination node v_m. The content in the elements is encrypted with the corresponding public keys.

5.3.2.2 *Message Generation*

When the source node generates a message, which has to be delivered to the destination within the TTL, it will create a contribution table and attach it onto the message. As shown in Figure 5.3, assume node v_1 is the source node that generates a message to the destination v_m, v_1 will create the contribution table including the header and elements. The header contains the information of the source node ID, destination node ID, TTL of the message and the generated time of the message (denoted as t_g), which is filled by the source node. Each of the elements will be added by each of the intermediate nodes when the message is forwarded.

5.3.2.3 *Message Forwarding*

When an intermediate node, take node v_i for example, forwards the message to its encounter v_{i+1}, it will add a new element E_i onto the contribution table of this message. E_i contains the information of the transmitter ID v_i, the receiver ID v_{i+1}, the time v_i receives the message and the time v_i transmits the message to v_{i+1}. To guarantee the security of the contribution table, v_i can encrypt the element using its public key PK_i, i.e., $E_i = Enc(PK_i, v_i \| v_{i+1} \| t_i^r \| t_i^t)$, where t_i^r denotes the time v_i receives the message and t_i^t denotes the time v_i transmits the message. It is assumed that all the nodes are time synchronized. Note that it must satisfy $t_i^t > t_i^r$ and $t_i^t = t_{i+1}^r$ when $i < m - 1$. As shown in Figure 5.3, when the source node v_1 meets v_2, it will transmit the message to v_2, before which it adds an element $E_1 = Enc(PK_1, v_1 \| v_2 \| t_1^r \| t_1^t)$ onto the contribution table, where t_1^r is the time v_1 generates the message, i.e., $t_1^r = t_g$, and t_1^t is the time v_1 transmits the message to v_2, i.e., $t_1^t = t_2^r$. To make sure this equation is satisfied, the transmitter can time-stamp the transmitting time onto the message after that the receiver can check whether this time is consistent with its current time clock (a certain deviation is allowed), and considers it as the receiving time.

5.3.2.4 Rewarding and Charging

After the last intermediate node, v_{m-1} in Figure 5.3, delivers the message to the destination, it will add an element $E_{m-1} = Enc(PK_{m-1}, v_{m-1}||v_m||t^r_{m-1}||t^t_{m-1})$ onto the contribution table and submit the whole contribution table as the report message to the TTP. After receiving the report message, the TTP will first decrypt it using the corresponding private keys. As shown in Figure 5.3, after receiving the report message, the TTP can obtain the source ID v_1 and the destination ID v_m from the header, which is not encrypted. Then the TTP can use the private key RK_1 corresponding to the source node v_1 to decrypt the first element in the table as $Dec(RK_1, Enc(PK_1, v_1||v_2||t^r_1||t^t_1)) = v_1||v_2||t^r_1||t^t_1$. As the first element contains the ID of next node v_2 in the path, the TTP knows the corresponding private key RK_2 of the second element and it can use RK_2 to decrypt the second element. Similarly, the TTP can decrypt the whole contribution table using a chain-decryption method.

To motivate the cooperation among the nodes in the message forwarding, the TTP will reward some credits to the nodes who contribute to the delivery of the message. The set of all the delivery paths of a message m_k is denoted as $\mathbb{P}^k$. In the EPSR scheme, only the nodes in the earliest delivery path P can get the reward for the delivery of m_k. P can be formulated as:

$$P = arg\ min\{t(P_i)|\forall P_i \in \mathbb{P}^k\}. \tag{5.3.3}$$

where $t(P_i)$ is the delivery time of path P_i. The reward to each node v_i in P is

$$Rd(v_i, m_k) = Ct(v_i, P), \tag{5.3.4}$$

where $Ct(v_i, P)$ is formulated in Eq. (5.3.1). Also, the rewarding credits, denoted as $Pay(S, m_k)$, will be paid by the source node as:

$$Pay(S, m_k) = \sum_{\forall v_i \in P} Rd(v_i, P). \tag{5.3.5}$$

As shown in Figure 5.2, the source node S generates a message m_k at 9:55am for the destination D and m_k is delivered along three paths to D. When the last intermediate nodes of the three paths, nodes C, G and J deliver m_k to D at 10:45am, 10:50am and 11:00am respectively, they will submit the contribution tables to the TTP. After decrypting the submitted contribution tables, the TTP can determine that the path $P = S \rightarrow A \rightarrow B \rightarrow C \rightarrow D$ is the earliest delivery path of m_k according to Eq. (5.3.3). As the contribution times of nodes A, B and C are $Ct(A, P) = 20$ minutes, $Ct(B, P) = 5$ minutes and $Ct(C, P) = 5$ minutes, according to Eq. (5.3.4), $Rd(A, m_k) = 20$, $Rd(B, m_k) = 5$ and $Rd(C, m_k) = 5$. Thus $Pay(S, m_k) = 30$. As S is the source node, it will not get the reward from itself.

5.3.3 Analysis

This section will analyze the proposed EPSR scheme from the three aspects: incentive compatibility, budget control and security enhancement.

5.3.3.1 Incentive Compatibility

A rewarding scheme is considered to be *incentive compatible* if the truthful cooperation (i.e., truthfully forward the message during each contact) is adopted by all the nodes, despite of their selfish nature. As shown in Figure 5.2, when node A meets node B at 10:35am, it forwards message m_k to B and finally the earliest delivery path $P = S \rightarrow A \rightarrow B \rightarrow C \rightarrow D$ and $Rd(A, m_k) = 20$. However, if A does not forward m_k to B at 10:35am, instead of which it chooses to forward m_k to H at 10:40am, then the earliest path will be $P = S \rightarrow A \rightarrow H \rightarrow I \rightarrow J \rightarrow D$ and $Rd(A, m_k) = Ct(A, P) = 10{:}40\text{am} - 10{:}15\text{am} = 25$. Thus, it is required to exam whether the EPSR scheme is incentive compatible, i.e. node A will truthfully forward m_k to node B at 10:35am, or not.

The incentive compatibility of the EPSR scheme relies on the *risk aversion* nature of individuals. The risk aversion refers to the reluctance of an individual to accept a bargain with an uncertain payoff rather than another bargain with a more certain, but possibly lower, expected payoff. The individual with risk aversion is apt to choose a strategy with a higher probability to get the payoff. In the noncooperative DTNs, whether a node is risk aversion or not depends on its current financial status. A node with a limited deposit tends to behave in a risk aversion way since it has to maximize the probability to get the reward to increase its deposit, which will be used to pay for the delivery of its own generated message. In the initialization phase of the proposed EPSR scheme, the system will allot a small amount of credits to each node, which can only afford the delivery of several messages. It is assumed that fabricating an identity requires a cost such as the registration fee, which is more than the allotted credits, so each node has no incentive to use a different identity each time to benefit from the allotted credits. Thus, during the routing procedure, each node will act in a risk aversion way.

Theorem 5.3.1. *The EPSR scheme is incentive compatible.*

Proof. As each node in the network will act in a risk aversion way, it will always take a strategy to maximize the probability of getting the reward. Assume that node v_i holds a message m_k, when it meets node v_j, it has two candidate strategies, i.e., forwarding m_k to v_j or not. If v_i forwards m_k to v_j, the probability that it is in the earliest delivery path of m_k will increase, indicating that v_i will have a higher probability to get the reward. However, if v_i refuses to forward m_k to v_j, the probability for v_i to be in the earliest path of m_k will decrease. As a risk-aversion node, v_i will decide to truthfully forward m_k to v_j to maximize its probability of getting the reward. Thus, the EPSR scheme is incentive compatible. □

5.3.3.2 Budget Control

In the EPSR scheme, for each delivered message, only the intermediate nodes in the earliest path can get the rewards according to their contribution time. Thus, the following theorem can be obtained.

Theorem 5.3.2. *The payment of the source node for each delivered message in the EPSR scheme is upper bounded.*

Proof. According to Eq. (5.3.2) and Eq. (5.3.5), the payment of the source node S for a delivered message m_k in the EPSR scheme is:

$$Pay(S, m_k) = \sum_{\forall v_i \in P} Rd(v_i, P) = \sum_{\forall v_i \in P} Ct(v_i, P) = TTL - t_r \leq TTL. \tag{5.3.6}$$

Eq. (5.3.6) suggests that the payment of the source node for each delivered message is upper bounded by the TTL of the message.

□

According to Theorem 5.3.2, it can be concluded that, under the scenario where each node has a finite budget, the proposed EPSR scheme is applicable.

Lemma 5.3.1. *It is deficit-free*[5.2] *for the TTP in the EPSR scheme.*

Proof. Since the source node only needs to pay the rewards to the nodes in the earliest delivery path, it is obvious that the payment is not less than the total rewards for any given transaction. Thus, it is deficit-free for the TTP. □

Lemma 5.3.1 can prevent the malicious node making extra profit from false transactions.

5.3.3.3 Security Enhancement

As the nodes in the network are selfish, they may cheat the system to get extra rewards by conducting malicious behaviors. To analyze the security issue of the EPSR scheme, it is assumed

[5.2] The deficit-free here means that the payment from the source node for any given transaction is no less than the amount of the total rewarding credits to the intermediate nodes.

that node v_i holds a message m_k while meeting with node v_j. This section will discuss if v_i will conduct the malicious behaviors when it forwards m_k to v_j.

Theorem 5.3.3. *In the EPSR scheme, node v_i has no incentive to launch the edge insertion attack, edge removal attack or content modification attack.*

Proof. Suppose P is the earliest delivery path of m_k. For the case that $v_i \notin P$ and $v_j \notin P$, launching one of the three attacks when it forwards m_k to v_j has no benefit for v_i as $Rd(v_i, m_k)$ is always 0. For the case that $v_i \in P$ and $v_j \notin P$, suppose path P' is another delivery path of m_k where $v_i \in P'$ and $v_j \in P'$. Then v_i also has no incentive to launch one of the three attacks since launching the attacks can only modify $Ct(v_i, P')$ but not $Ct(v_i, P)$, indicating that $Rd(v_i, m_k)$ will not be changed by launching the attacks. For the case that $v_i \in P$ and $v_j \in P$, the edge insertion attack, edge removal attack and content modification attack will be considered respectively.

1) Edge insertion attack: As $v_i \in P$, $Rd(v_i, m_k) = Ct(v_i, P)$. Without loss of generality, if node v_i inserts an edge $\overrightarrow{v_i v_{i+1}}$ (v_{i+1} is the sybil of v_i) to the path P, then both v_i and v_{i+1} will get the rewards from the TTP. However, $Ct'(v_i, P) + Ct'(v_{i+1}, P) = Ct(v_i, P)$, where $Ct'(\cdot)$ indicates the contribution time after inserting the edge. Thus, v_i cannot make extra profit by inserting an edge into the path. Consequently, node v_i has no incentive to launch an edge insertion attack.

2) Edge removal attack: If v_i launches an edge removal attack, i.e., v_i removes one of the elements from the contribution table, it will break the encryption chain of the contribution table. Then the contribution table cannot be totally decrypted after submitted to the TTP, and there will be no reward paid to the nodes. Thus, launching an edge removal attack will make $Rd(v_i, m_k)$ become 0. So v_i has no incentive to launch an edge removal attack.

3) Content modification attack: As $Rd(v_i, m_k) = Ct(v_i, P)$, v_i may modify the content of its contribution element, such as decreasing t_i^r or increasing t_i^t, to magnify $Ct(v_i, P)$. However, such modifications can be detected. Suppose the parent node of v_i is v_k, then in the contribution table, it must satisfy $t_i^r = t_k^t$ and $t_i^t = t_j^r$. Moreover, v_i cannot modify t_k^t or t_j^r for it does not have

the private keys of v_k and v_j. Thus, the modification of t_i^r and t_i^t can be easily detected by the TTP, making v_i get no reward. So v_i has no incentive to launch a content modification attack.

Thus, the EPSR scheme can prevent the nodes from launching the edge insertion attack, edge removal attack, content modification attack, or the combination of them. □

5.4 Earliest Path Cumulative Rewarding Scheme

In the EPSR scheme, it is incentive compatible for each node to truthfully forward the messages. However, a node may be aware of more contact information of other nodes, under which it may not truthfully forward the message. For example, in Figure 5.2, there are three paths from the source to destination. Node A receives message m_k from S at 10:15am and then it meets B at 10:35am. If A does not know the contact information of other nodes, it will truthfully forward m_k to B to increase its probability to be in the earliest path. Thus, $Rd(A, m_k) = 20$. However, if A knows that there are only three paths from the source to destination, all going through itself, it may not forward m_k to B at 10:35am. Instead it can forward m_k to H at 10:40am and the final earliest path will be $S \rightarrow A \rightarrow H \rightarrow I \rightarrow J \rightarrow D$. In this way, $Rd(A, m_k) = 25$, which is more than 20. Thus, the EPSR scheme may be not applicable under this kind of scenario. The earliest path cumulative rewarding (EPCR) scheme will be proposed to solve this problem.

5.4.1 EPCR Scheme

The main idea of the EPCR scheme is to reward each node in the earliest delivery path in a cumulative way. The reward to each node in the earliest path includes two parts, the first part is its contribution time in the earliest delivery path and the second part is its contribution time in other delivery paths. The second part of reward to the node can guarantee that it will truthfully forward the message to its encounter during every contact even when it is aware of the contact information of other nodes. The rewards for the delivery of the message will be paid by the source node.

Similar to EPSR, the EPCR scheme includes four phases: system initialization, message generation, message forwarding, and rewarding and charging. The main difference between the EPCR and EPSR is in the fourth phase.

Before introducing the detail of the rewarding mechanism, the concept of the *furcation node* of two paths will be first introduced: A furcation node of two paths is the last intersection node (except the destination node) of the two paths. The furcation node v_i of paths P_j and P_k, denoted as $Fn(P_j, P_k)$, is defined that if there exist two edges e_l and e_m such that $v_i \in e_l, e_l \in P_j, e_l \notin P_k$ and $v_j \in e_m, e_m \in P_k, e_m \notin P_j$, then $Fn(P_j, P_k) = v_i$. For example, in Figure 5.2, node A is the furcation node of path $S \rightarrow A \rightarrow B \rightarrow C \rightarrow D$ and path $S \rightarrow A \rightarrow H \rightarrow I \rightarrow J \rightarrow D$. Note that $Fn(P_j, P_k) = \emptyset$ if $j = k$.

In the rewarding and charging phase, only the nodes in the earliest delivery path can get the rewards from the TTP. However, the reward to each node is different with that of the EPSR scheme. Suppose the set of all the delivery paths of a message m_k is $\mathbb{P}^k = \{P_1, P_2, \cdots, P_l\}$ and P is the earliest delivery path of m_k, i.e., $P = arg\ min\{t(P_i) | \forall P_i \in \mathbb{P}^k\}$. Then, the reward to each node v_i in P for the delivery of m_k is

$$Rd(v_i, m_k) = Ct(v_i, P) + \sum_{j=1}^{l} [w(v_i, P_j) \cdot Ct(v_i, P_j)] \tag{5.4.1}$$

where

$$w(v_i, P_j) = \begin{cases} 1, & \text{if } v_i = Fn(P, P_j), \forall m \in [1, j), v_i \neq Fn(P, P_m); \\ 0, & \text{elsewhere.} \end{cases}$$

Also, the rewarding credits for the delivery of m_k will be paid by the source node denoted as $Pay(S, m_k)$, where

$$Pay(S, m_k) = \sum_{\forall v_i \in P} Rd(v_i, P). \tag{5.4.2}$$

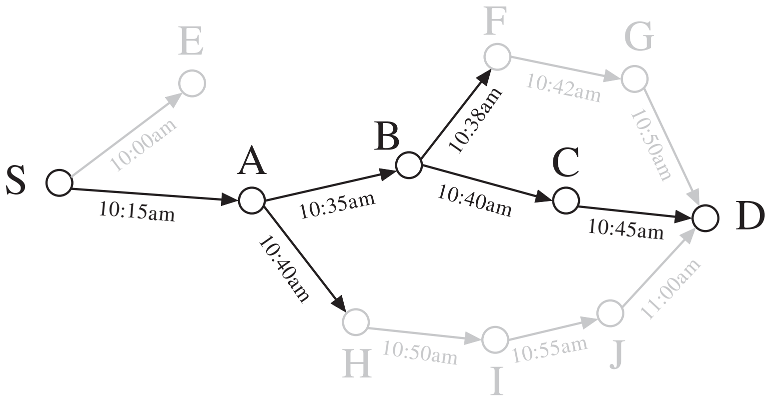


Figure 5.4. Illustration of the total contribution time to be rewarded for the example in Figure 5.2 under the earliest-path cumulative rewarding scheme. The gray parts will be ignored in the calculation of the rewarded contribution time.

Take Figure 5.2 for example, after the delivery of m_k, the nodes A, B and C will get rewards from the TTP. According to Eq. (5.4.1), $Rd(A, m_k)$=(10:35am-10:15am)+(10:40am-10:15am) = 45, $Rd(B, m_k)$=(10:40am-10:35am)+(10:38am-10:35am) = 8 and $Rd(C, m_k)$=(10:45am-10:40am) = 5 as shown in Figure 5.4. The EPCR scheme can solve the problem in the previous scenario as follows: if A does not forward m_k to B at 10:35am and waits until 10:40am to forward m_k to H, then the path $S \rightarrow A \rightarrow H \rightarrow I \rightarrow J \rightarrow D$ will be the earliest delivery path and $Rd(A, m_k) = 25$. However, if A truthfully forwards m_k to each of its encounters, then $Rd(A, m_k) = 45$. Thus, A will truthfully forward the message during every contact opportunity.

5.4.2 Analysis

5.4.2.1 Incentive Compatibility

In the initialization phase of the EPCR scheme, the system will provide each node with a small amount of credits, which can only afford the cost for delivering several messages. Thus each node will act in a risk aversion way to maximize its probability to get the reward.

Theorem 5.4.1. *The EPCR scheme is incentive compatible.*

Proof. As each node in the network will act in a risk aversion way, it will always adopt a strategy to maximize its probability to get the reward. Suppose that node v_i holds a message m_k, when it meets node v_j, it has two candidate strategies, i.e., v_i forwards m_k to v_j or not. If v_i forwards m_k to v_j, its probability to be in the earliest path of m_k will increase, indicating that v_i will have a higher probability to get the reward. Moreover, the forwarding of m_k to v_j may increase $Rd(v_i, m_k)$ since it may make v_i be the furcation node between the earliest delivery path and another path. However, if v_i does not forward m_k to v_j, its probability to be in the earliest path of m_k will decrease, indicating a lower probability to get the reward. As a risk-aversion node, v_i will decide to truthfully forward m_k to v_j. Thus, the EPCR scheme is incentive compatible. □

Theorem 5.4.1 can guarantee that the dominant strategy for the nodes is to truthfully forward the messages during every contact opportunity.

5.4.2.2 Budget Control

In DTNs, the transmission of a message requires some time period due to the limited transmission bit rate. The time period to transmit a unified-size message is denoted as T. Thus, the upper bound of the number of nodes in a delivery path is $N = \lfloor \frac{TTL}{T} \rfloor$ (it is only considered that the number of the transmitters, i.e., N does not include the destination node).

Theorem 5.4.2. *The maximum payment of the source node for a delivered message m_k in the EPCR scheme is:*

$$Pay(S, m_k)^{max} = \begin{cases} f(min\{\frac{N+1}{2}, n-2\}) \cdot T, & \textit{if N is odd;} \\ f(min\{\frac{N}{2}+1, n-2\}) \cdot T, & \textit{elsewhere.} \end{cases}$$

where $f(x) = -x^2 + (N+3)x - N - 1$, $N = \lfloor \frac{TTL}{T} \rfloor$ and n is the number of nodes in the network.

Proof. Suppose that message m_k is generated by source node S at t_0 and S immediately transmits m_k to node v_1. After that, m_k is forwarded along nodes v_1, v_2, $\cdots$, v_{n_1}, and then to the destination node D, and the path $S \rightarrow v_1 \rightarrow v_2 \rightarrow \cdots \rightarrow v_{n_1} \rightarrow D$ is the earliest delivery path. Then, the case for S to pay the maximum credits is shown in Figure 5.5, that is, v_{n_1-1} will transmit m_k directly to D at time t_{N-1}, and v_{n_1-2} will transmit m_k directly to D at time t_{N-2}, and so on.

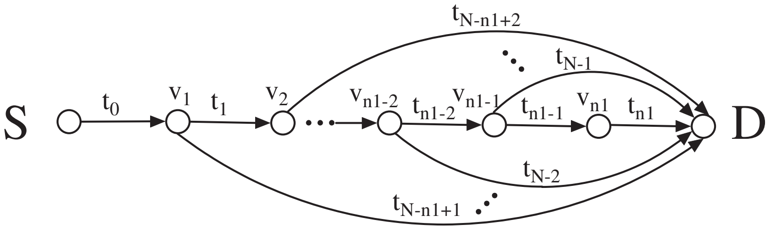


Figure 5.5. The calculation of the maximum payment for the EPCR scheme. The time on each arrow denotes the transmitting time and $t_j - t_i = (j - i) \cdot T$, where $j > i$.

According to Eq. (5.4.1), after the delivery of m_k, v_1 can get a reward of $(t_1 - t_0) + (t_{N-n_1+1} - t_0)$. Note that the last intermediate node v_{n_1} can only get a reward of $(t_{n_1} - t_{n_1-1})$ as it only has the contribution time in the earliest path. According to Eq. (5.4.2), the payment of S is:

$$\begin{aligned}
Pay(S, m_k) &= [(t_1 - t_0) + (t_{N-n_1+1} - t_0)] + [(t_2 - t_1) + (t_{N-n_1+2} - t_1)] \\
&\quad + \cdots + [(t_{n_1-1} - t_{n_1-2}) + (t_{N-2} - t_{n_1-1})] + (t_{n_1} - t_{n_1-1}) \\
&= (t_{n_1} - t_0) + [(t_{N-n_1+1} - t_0) + (t_{N-n_1+2} - t_1) + \cdots + (t_{n_1-1} - t_{n_1-2})] \\
&= (t_{n_1} - t_0) + \sum_{i=1}^{n_1-1} (t_{N-n_1+i} - t_{i-1}) = n_1 \cdot T + (n_1 - 1)(N - n_1 + 1) \cdot T \\
&= [-n_1^2 + (N + 3)n_1 - N - 1] \cdot T.
\end{aligned}$$

It is obvious that $Pay(S, m_k)$ is an increasing function of n_1 when $n_1 < (N + 3)/2$.

As the path $S \rightarrow v_1 \rightarrow v_2 \rightarrow \cdots \rightarrow v_{n_1} \rightarrow D$ is the earliest delivery path of m_k, the moment that node v_{n_1} transmits m_k to D is at least T earlier than other moments that m_k is transmitted to D by other nodes. Thus, it can be obtained that $t_{n_1} \leq t_{N-n_1+1} + T$, i.e., $n_1 \leq N - n_1 + 2$. So n_1 must satisfy $n_1 \leq N/2 + 1$. Also, as the number of nodes in the network is n, it can be obtained that $n_1 + 2 \leq n$ (n_1 is the number of nodes in the earliest path except the source and destination).

Thus:

$$Pay(S, m_k) = [-n_1^2 + (N+3)n_1 - N - 1] \cdot T,$$

where $n_1 \leq N/2 + 1$ and $n_1 \leq n - 2$.

So if N is an odd number, $Pay(S, m_k)$ will be the largest when $n_1 = min\{(N-1)/2+1,\ n-2\}$. Otherwise, if N is an even number, $Pay(S, m_k)$ will be the largest when $n_1 = min\{N/2+1,\ n-2\}$. Thus,

$$Pay(S, m_k)^{max} = \begin{cases} f(min\{\frac{N+1}{2}, n-2\}) \cdot T, & \text{if N is odd;} \\ f(min\{\frac{N}{2}+1, n-2\}) \cdot T, & \text{elsewhere.} \end{cases} \tag{5.4.3}$$

□

Lemma 5.4.1. *The payment of the source node for each delivered message in the EPCR scheme is upper bounded.*

Proof. The maximum payment of the source node for a delivered message m_k in the EPCR scheme is formulated in Eq. (5.4.3). Since the function $f(\cdot)$ in Eq. (5.4.3) is continuous and all the parameters N, TTL and n are finite, it is obvious that the payment of the source node for each delivered message in the EPCR scheme is upper bounded. □

According to Lemma 5.4.1, it can be concluded that the EPCR scheme is applicable under the scenario that each node has a finite budget. Although the bound of the payment of the source node for each delivered message may seem large, it can be argued that it is feasible in practice since the reward to each node for the delivered message also increases.

Lemma 5.4.2. *It is deficit-free for the TTP in the EPCR scheme.*

Proof. Since the payment of the source node for each delivered message equals to the total rewards paid to the nodes in the earliest delivery path of the message, the EPCR scheme is budget balance. Thus, it is deficit-free for the TTP. □

Lemma 5.4.2 guarantees the property of deficit-free of the EPCR scheme, which can prevent the malicious node from making profit by phantom transactions.

5.4.2.3 Security Enhancement

Due to the natural selfishness, each node may launch one of the three attacks to get more reward. This section will discuss the security issue of the EPCR scheme. Similarly, it is assumed that node v_i holds a message m_k while meeting v_j. Then whether v_i will conduct the attacks when it forwards m_k to v_j will be discussed.

Theorem 5.4.3. *In the EPCR scheme, node v_i has no incentive to launch the edge insertion attack, edge removal attack or content modification attack.*

Proof. Suppose P is the earliest delivery path of m_k. For the case that $v_i \notin P$ and $v_j \notin P$, v_i has no incentive to launch one of the three attacks before forwarding m_k to v_j as $Rd(v_i, m_k)$ is still 0. For the other cases that $v_i \in P$ and $v_j \in P$, or $v_i \in P$ and $v_j \notin P$, the edge insertion attack, edge removal attack and content modification attack will be considered respectively.

1) Edge insertion attack: If $v_i \in P$ and $v_j \notin P$, then v_i can get the reward according to Eq. (5.4.1). For instance, suppose $v_j \in P'$ where P' is not the earliest path, then if v_i inserts an edge between itself and v_j such as $\overrightarrow{v_i v_{i+1}}$, then P' will contain $\overrightarrow{v_i v_{i+1}}$ and $\overrightarrow{v_{i+1} v_j}$, and $Rd(v_{i+1}, m_k) = 0$ since $v_{i+1} \notin P$. Consequently, $Ct(v_i, P')$ will be reduced by $t^t_{i+1} - t^r_{i+1}$ as the time interval between v_i and v_j in path P' cannot be modified, where t^r_{i+1} is the time that v_{i+1} receives m_k from v_i and t^t_{i+1} is the time that v_{i+1} transmits m_k to v_j. Note that t^r_{i+1} and t^t_{i+1} are actually determined by v_i and they must satisfy $t^r_i < t^r_{i+1} < t^r_j$ and $t^t_{i+1} = t^r_j$. Thus, launching an edge insertion attack will reduce $Rd(v_i, m_k)$ (since v_i is the furcation node between P and P', thus $Ct(v_i, P')$ will be included in $Rd(v_i, m_k)$), leading it has no incentive to do that. On the other case that $v_i \in P$ and $v_j \in P$, if v_i inserts an edge between itself and v_j, similarly assume v_i inserts a sybil node v_{i+1} between itself and v_j, then $Ct'(v_i, P) + Ct'(v_{i+1}, P) = Ct(v_i, P)$, where

$Ct'(\cdot)$ denotes the contribution time after inserting the edge. Then v_i cannot increase $Rd(v_i, P)$ after inserting the edge. Thus, v_i has no incentive to launch an edge insertion attack.

2) Edge removal attack: Under both the two cases, i.e., $v_i \in P$ and $v_j \notin P$, or $v_i \in P$ and $v_j \in P$, if v_i launches an edge removal attack, then after the contribution table is submitted to the TTP, it cannot decrypt the whole contribution table as the removal of the contribution element breaks the decryption chain of the contribution table. Therefore, the TTP will not reward the credits to the nodes since the contribution table can not be correctly decrypted. Thus, launching an edge removal attack will lead $Rd(v - i, m_k) = 0$, which makes it have no incentive to launch an edge removal attack.

3) Content modification attack: Under both the two cases that $v_i \in P$ and $v_j \notin P$, or $v_i \in P$ and $v_j \in P$, v_i may modify the content of its contribution element, such as decreasing t_i^r or increasing t_i^t, to magnify $Ct(v_i, P)$, through which it increase $Rd(v_i, m_k)$. However, suppose that the parent node of v_i is v_k, then in the contribution table, it must satisfy that $t_i^r = t_k^t$ and $t_i^t = t_j^r$. Moreover, v_i can not modify t_k^t or t_j^r since it does not have the private keys of v_k and v_j. Thus, the modification of the contribution time of only t_i^r and t_i^t can be easily detected by the TTP, making v_i get no reward. Thus v_i has no incentive to launch a content modification attack. □

According to Theorem 5.4.3, the EPCR scheme can prevent the nodes launching the edge insertion attack, edge removal attack or content modification attack (or the combination of them).

5.5 Performance Evaluation

This section evaluates the proposed rewarding schemes in the Opportunistic Network Environment simulator (ONE) [33] using the UMassDieselNet trace [5]. The objectives are threefold: 1) to illustrate the necessity of the rewarding schemes to motivate the cooperation in the message forwarding; 2) to validate the feasibility of the proposed rewarding schemes; 3) to analyze

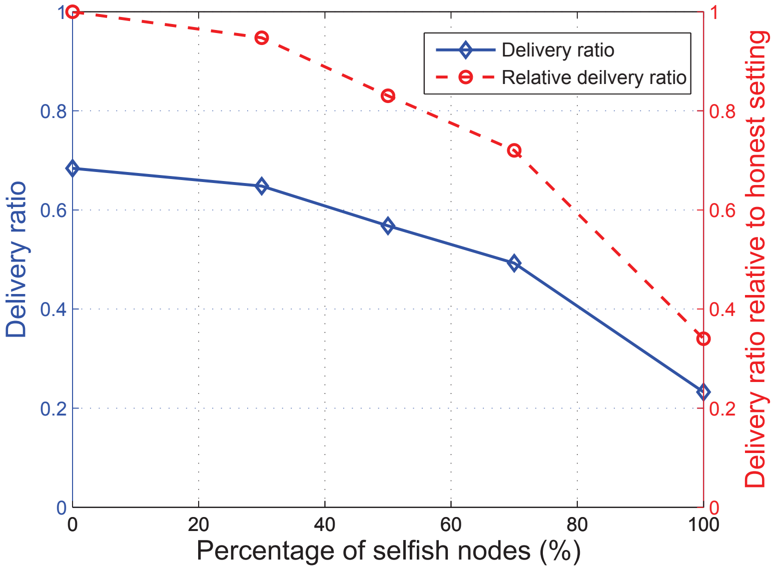


Figure 5.6. Delivery ratio and relative delivery ratio under different percentage of selfish nodes.

the effects of the network parameters on the proposed rewarding schemes.

5.5.1 Performance Metrics and Simulation Settings

The performance will be evaluated with the four metrics, i.e., (relative) delivery ratio, (relative) latency, total rewarding credits and average rewarding credits. The parameters in the simulations are set as follows: the size of each message is 1 KB and the buffer space of each node is 1 GB. The transmission rate is 2 Mbps. The default number of generated messages during the whole period of simulation is 1500 and the default time-to-live (TTL) of each message is 3000 minutes. In the simulations, one unit of the rewarding credit corresponds to one second of the contribution time. The value of each point in the curves is the average of 10 simulation runs.

5.5.2 Simulation Results

In Figure 5.6 and 5.7, the impacts of selfish nodes on the DTN routing performance are analyzed under the scenario that no rewarding scheme is adopted, in which some percentage of nodes will behave selfishly. In the simulations, a selfish node will act in a free-riding way such that it

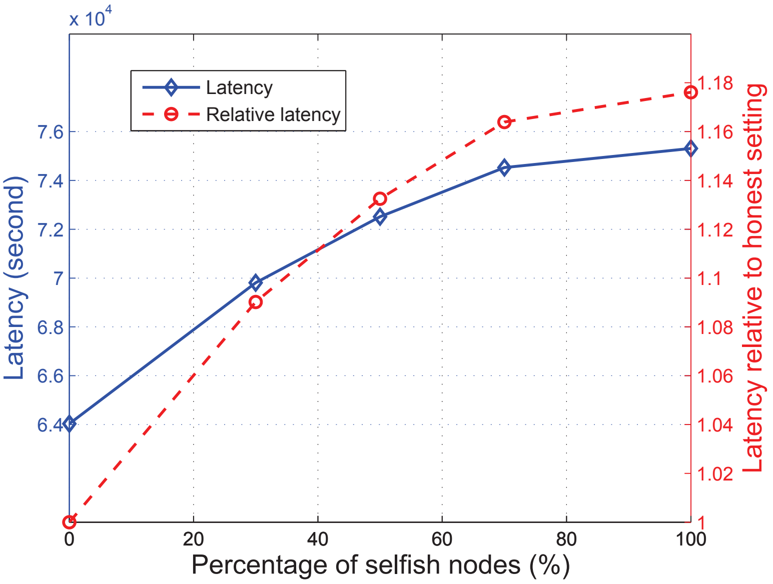


Figure 5.7. Latency and relative latency under different percentage of selfish nodes.

will only forward its own generated messages but decline to forward those generated by other nodes. In these two figures, the number of generated messages is set as 1500 and the TTL of each message is set as 3000 minutes. Figure 5.6 shows the effects of the selfish nodes on the delivery ratio and relative delivery ratio. The solid-line curve in Figure 5.6 depicts the decrease trend of the delivery ratio when the percentage of selfish nodes increases. When all the nodes are non-selfish, i.e., the percentage of selfish nodes equals to 0, then the delivery ratio can be almost 70%, which is high enough for many practical applications. While when all the nodes are selfish, i.e., the percentage of selfish nodes equals to 100%, the delivery ratio drops dramatically to only 22%. Note that the delivery ratio with 100% selfish nodes can be larger than 0 since each source node may have a chance to directly deliver its own generated message to the destination. The dashed curve in Figure 5.6 depicts the decrease trend of the delivery ratio relative to that with no selfish node. It clearly shows the downgrade percentage of the delivery ratio when more selfish nodes exist.

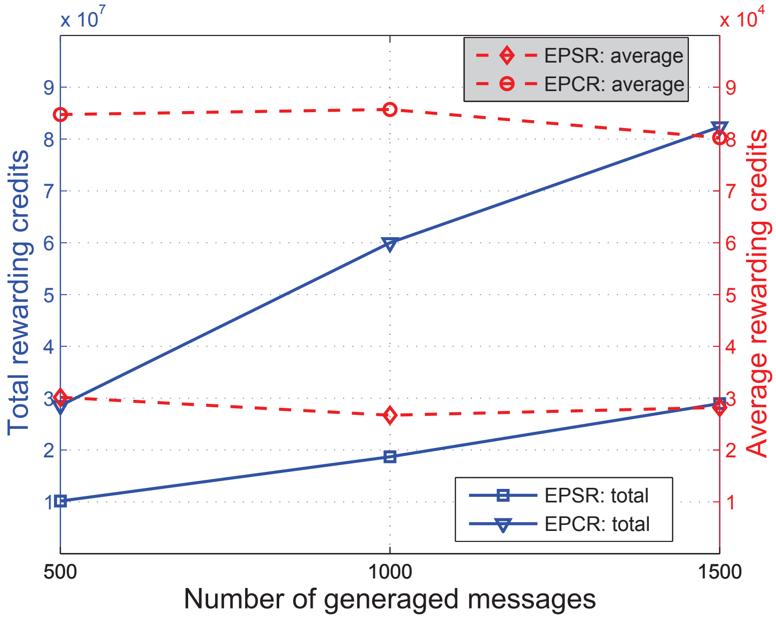


Figure 5.8. The total rewarding credits and average rewarding credits of the two proposed rewarding schemes under different number of generated messages.

Figure 5.7 shows the effects of the selfish nodes on the latency and relative latency. The solid-line curve reflects the increase trend of the latency when the percentage of selfish nodes increases. It shows that the latency is only 64,000 seconds when all the nodes are non-selfish. When the percentage of selfish nodes increases, the latency will rise dramatically. The latency will reach 75,300 seconds under the extreme case that all the nodes are selfish. The dashed curve clearly shows the increase percentage of the latency with different percentage of selfish nodes. It is illustrated that the latency will increase to 118% of that with no selfish node. As shown in Figure 5.6 and 5.7, the existence of the selfish nodes will dramatically decrease the delivery ratio and increase the latency. Thus, the effects of the selfish nodes on the message forwarding cannot be ignored and it is highly necessary to propose the rewarding schemes to motivate the cooperation in the message forwarding.

As mentioned in the above content, when the proposed rewarding schemes are adopted, all the nodes in the network will truthfully forward the messages. Figure 5.8 and 5.9 shows

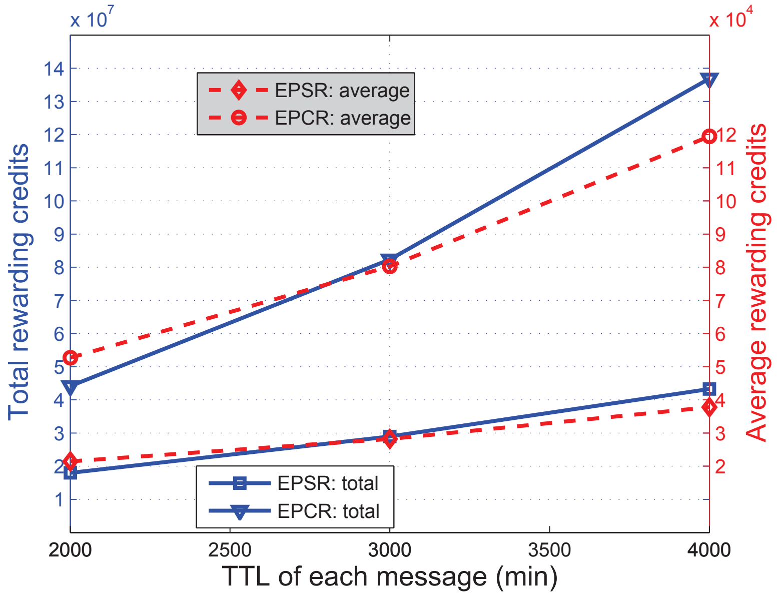


Figure 5.9. The total rewarding credits and average rewarding credits of the two proposed rewarding schemes under different TTL of each message.

the effects of the network parameters on the proposed two rewarding schemes. Figure 5.8 shows the total rewarding credits and average rewarding credits of the proposed two rewarding schemes under different number of generated messages (from 500 to 1500) respectively. Here the total rewarding credits mean the summation of the rewarding credits of all the delivered messages in the network and the average rewarding credits mean the average rewarding credits for each delivered message, i.e., the average rewarding credits = the total rewarding credits / the number of delivered messages. The TTL of each message in this figure is set as 3000 minutes. The solid-line curves depict the total rewarding credits of the two rewarding schemes, which increase almost linearly with the number of generated messages. Since the increase of generated messages will increase the number of delivered messages, resulting in the linear increase of the total rewarding credits. As the EPCR scheme rewards each node in the earliest delivery path in a cumulative manner, the total rewarding credits are more than that of the EPSR scheme. The dashed curves depict the average rewarding credits for each delivered message of the two rewarding schemes respectively. Similarly, the average rewarding credits of the EPCR scheme are more than that of the EPSR scheme. It shows hat the average rewarding credits of the two proposed rewarding schemes are quite stable under different number of generated messages, which validates the previous claim that the payment of the source node for each delivered message is upper bounded (as the rewarding credits are charged from the source node).

Figure 5.9 shows the effects of the time-to-live (TTL) of each message, which varies from 2000 minutes to 4000 minutes with the interval of 1000 minutes, on the total rewarding credits and average rewarding credits of the proposed two rewarding schemes respectively. The meanings of the total rewarding credits and average rewarding credits are the same as that in Figure 5.8. The solid-line curves illustrate the effects of the TTL on the total rewarding credits of the two rewarding schemes. The increase of the TTL will increase the total rewarding credits of the two rewarding schemes. As the rewarding credits of the rewarding schemes are based on the contribution time of the nodes, the increase of the TTL may increase the total contribution time of the nodes in each delivery path and the number of delivered messages will also increase with a longer TTL, which will result in more total rewarding credits. Figure 5.9 shows that the total rewarding credits of the EPCR scheme are more than that of the EPSR scheme. The dashed curves illustrate the effect of the TTL on the average rewarding credits for each delivered message. It shows that the average credits of the two rewarding schemes increase almost linearly with the TTL. Similarly, the average rewarding credits of the EPCR scheme are more than that of the EPSR scheme.

5.6 Summary

By considering the drawbacks of the previous incentive schemes, this chapter proposes the EPSR and EPCR schemes respectively to motivate the nodes to cooperate with each other in the message forwarding in the non-cooperative DTNs. The proposed rewarding schemes are proved to be incentive compatible, which guarantees that the dominant strategy for the nodes is to truthfully forward the messages. It is also proved that the payment for each delivered message is upper bounded, making the proposed rewarding schemes applicable for the scenario when the nodes have a finite budget. Furthermore, the proposed rewarding scheme are resistant to the malicious behaviors of the selfish nodes. The simulations based on the real trace are conducted to analyze the effects of the selfish nodes on the routing performance and illustrate the effectiveness of the proposed rewarding schemes.

CHAPTER 6

END-TO-END LOCATION PRIVACY PROTECTION IN DELAY TOLERANT EVENT COLLECTION SYSTEMS

6.1 Introduction

Wireless sensor networks (WSNs) [1] are made up of a number of sensor nodes that are self-organized to carry out various monitoring tasks such as soil monitoring [82], earthquake monitoring [51], and habitat monitoring [55]. In the habitat monitoring sensor networks, sensor nodes are deployed in the field to determine the occurrence of an interesting event such as the presence of a rare animal. A node who detects the occurrence of an event can send the information to a sink or base station for further analysis. This procedure is a kind of event collection. In the event collection, if the interesting event is delay tolerant, i.e., the sensor node only needs to send the monitoring information to the sink or base station within a certain delay constraint (but not immediately), it is called the delay tolerant event collection, which is one of typical applications of delay tolerant networks (DTNs).

In the delay tolerant event collection systems [90, 24], the sensor node who detects the occurrence of an event will send the information to the sink or base station via multi-hop wireless communications. As the wireless channels can be accessed by anyone who wishes, it is not difficult to attack wireless networks with the goal of either obtaining confidential data or simply disrupting the normal operation of the event collection systems. In either case, they may involve threats to one of two types of privacy, *content* privacy and *contextual* privacy [31]. The content privacy refers to the confidentiality of the content of the messages passing between the nodes in the network. This is usually guaranteed by using methods of encryption and authentication [17]. The contextual privacy refers to the confidentiality of information about traffic patterns in the network, which attackers may use to disrupt the network.

To illustrate how information about traffic patterns in a delay tolerant event collection system might be exploited by an adversary, this chapter considers the scenario of "panda-hunter" [31] in Figure 6.1, which shows a typical habitat monitoring sensor network where sensor nodes monitor the pandas in the environment and then send report messages to a sink by multi-hop wireless communications. There is a central controller (sink in Figure 6.1) and several pandas in the monitoring field. The sensor nodes, which detect the existence of the pandas, will act as the source nodes and they will report the monitoring information to the central controller via multi-hop wireless communications periodically. As the monitoring information collected from the sensor nodes will be analyzed in the sink, the source nodes can send the information to the sink within some delay constraint, i.e., it is a kind of delay tolerant event collection. The scenario is obvious unsafe as the hunter (adversary in Figure 6.1) is easily able to either locate a source by back tracing hop-by-hop to capture the panda or locate a receiver by following the flow of packets in the network to destroy the central controller, which will make the whole system crash. The challenge in the scenario is essentially to protect the end-to-end location privacy rather than merely the source or sink location privacy. Thus, the end-to-end location privacy protection is one of the most important contextual privacy problems in the delay tolerant event collection systems.

Lots of location privacy protection schemes for the routing in the delay tolerant event collection systems have been developed in the past decade. However, these proposed schemes can only protect the source-location privacy or the sink-location privacy independently. This chapter proposes four end-to-end location privacy protection schemes to protect against a local eavesdropper who might breach the location privacy of a source or sink, that is, end-to-end location privacy. The four schemes are forward random walk (FRW), bidirectional tree (BT), dynamic bidirectional tree (DBT) and zigzag bidirectional tree (ZBT) respectively. In the for-

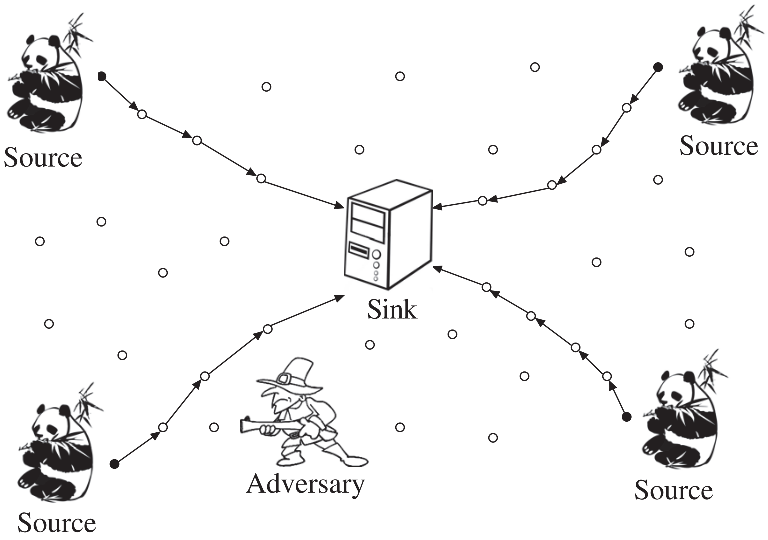


Figure 6.1. Location privacy sensitive scenario in the habitat monitoring sensor network.

ward random walk scheme, every node relays a received message to a node randomly chosen from its forward neighbors whose hop-count to the sink is no larger than its own. This procedure is repeated at each node until the message arrives at the sink. To increase the location anonymity, tree topology is employed at the two ends of the routing path respectively in the bidirectional tree scheme. In the dynamic bidirectional tree scheme, branches of the trees are generated dynamically, which can improve the performance. However, in the bidirectional tree scheme, real messages are routed along the shortest path, which makes it possible for the eavesdropper to infer the location of the source and sink by extending the line of the shortest path. To solve this potential threat, a proxy source and a proxy sink are devised in the zigzag bidirectional tree scheme, making it more difficult for the adversary to obtain the location of the source or sink.

6.2 System Scenario, Adversary Model and Location Privacy Metrics

This section will describe a generic scenario, in which a habitat monitoring sensor network is potentially threatened by a particular adversary, where the adversary seeks to breach the location privacy of the source or sink in the network. After that the adversary model will be detailedly introduced. Finally, the metrics safety period, latency and energy consumption will be defined to evaluate the location privacy of the routing schemes.

6.2.1 System Scenario

This chapter considers an event collection system based on the habitat monitoring sensor network that is comprised of a sink node and many sensor nodes. The sensor nodes, which detect the existence of the animal, will send the monitoring information to the sink via multi-hop wireless communications. As the monitoring information collected from the sensor nodes will be analyzed in the sink, the sensor nodes can send the information to the sink within some delay constraint. Thus it is a delay tolerant event collection system, which is one of the typical DTN applications. The goal in this chapter is to protect the location privacy of both the sources and sink in such a delay tolerant event collection system.

It is assumed that all the sensor nodes are identically configured. Thus they have the same capability and communication range R. Two sensor nodes can communicate with each other when their distance is less than R. The sink is assumed to have greater capabilities than the sensor nodes. During the initialization phase of the network, the sink originates a flooding, which provides each sensor node with three kinds of information: 1) the least hop count from itself to the sink; 2) its neighboring nodes; 3) the least hop count from each neighboring node to the sink. During the report period T_r, each source sends a message to the sink periodically with an interval of T_S using certain routing strategies. The location privacy of the sources and sink are regarded as intact if none of them can be identified by the adversary within the report period T_r.

6.2.2 Adversary Model

It is assumed that the attackers are equipped with some powerful devices such as spectrum analyzer, which can be used to locate the sender of a transmitted message [28]. The attackers can also move freely in the network. Typically, the attackers against the contextual privacy can be classified into two categories: *local* (or *mote-class*) attackers and *global* (or *laptop-class*) attackers [32]. Local attackers are assumed to have a local view of the network traffic, which means that they can only eavesdrop on the messages within the transmission range. Global attackers acquire a global view of the network traffic, based on which they can intercept every message in the network. The global attackers have serious effects on the network but they are very difficult to implement, especially in a large scale wireless sensor network. This chapter focuses on the location privacy protection of both the source and sink against the local adversary.

It is assumed that each message transmitted in the network is encrypted and the adversary cannot access the content. Thus, a local adversary is reduced to identifying either the source (or the sink) by analyzing the traffic flow and tracing back (or forth) hop-by-hop. The characteristics of a local adversary are defined as follows, some of which are borrowed from the "panda-hunter" model [31]:

- The adversary randomly walks in the network until it eavesdrops a message transmitted by some sensor node. The adversary then randomly decides whether to trace the source or sink.

- The adversary is equipped with some powerful devices, such as antenna and spectrum analyzer, which can be used to measure the arriving angle and the received signal strength of a message. Based on the above two measurements, the adversary can identify the location of the immediate sender.

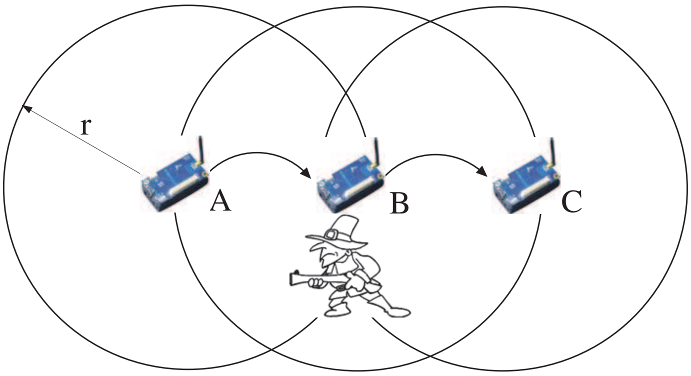


Figure 6.2. The adversary eavesdrops on messages in the network.

- The adversary is able to detect the existence of the target (the source or sink) when it is close enough.
- The movement of the adversary is far slower than the transmitting speed of a message in the network. Therefore, the adversary can only trace the flow by one hop for one packet transmission.
- The adversary will not actively interfere with the message transmission in the network as there may exist intrusion detection mechanisms.
- The adversary has enough memory space to save the trace information, and if it receives no more message for a long time, it may retreat to a previously visited location.
- According to Kerckhoff's Principle, the adversary is aware of the routing strategies of the network.

An adversary can initially move around and wait for eavesdropping a message. As soon as it detects a new message, it can determine the location of the immediate sender for tracing the source. It can then move to that location and wait there for the next message. To trace the sink, the adversary needs to identify the direction of the message and then moves to the receiver of the message. In Figure 6.2 for example, the adversary stays at node B. If the adversary wants to trace the source, it will move to node A as soon as node A transmits a message to node B. On the other hand, if the adversary wants to trace the sink, it can identify the direction of the message as follows: It detects a message transmitted from node A when node A sends a message to node B. Shortly after that, node B transmits a message and soon again after node C transmits a message. The adversary identifies the transmission sequence A→B→C and node C is the last receiver. The adversary then moves to node C.

This chapter assumes that the adversary behaves according to one of two models: the patient adversary model and the cautious adversary model. In both models, the adversary first randomly walks and detects messages in the network. As soon as it detects a message, it triggers a hop-by-hop tracing procedure to capture either the source or the sink. In the patient adversary model, the adversary will use the above technique patiently until it captures its target, i.e., the source or sink. In the cautious adversary model, the adversary will trace back if it waits for a given period at some location. The path that the adversary visited is denoted as $\mathcal{L} = \{l_1, l_2, ..., l_{k-1}, l_k\}$, where l_k is the current location of the adversary. When the adversary has not received any new message within a specific interval at l_k, it will trace back along $\mathcal{L}$ to l_{k-1}, delete l_k in $\mathcal{L}$ and then wait there for new message. The set of locations that the adversary has visited and traced back is denoted as $\mathcal{F}$. To avoid invalid tracing, when the adversary traces back from l_k to l_{k-1}, it will add l_k into $\mathcal{F}$, and ignore messages coming from any location in $\mathcal{F}$. Also, the adversary can avoid getting lost in a loop with loop detection techniques.

6.2.3 Metrics of Location Privacy

The proposed end-to-end location privacy protection schemes are evaluated using three metrics: the safety period, latency, and energy consumption.

- *Safety period:* The safety period begins from the moment that the adversary triggers the tracing procedure (i.e., detects the first message) and ends at the moment when the adversary identifies the source or sink. It is measured using the ratio of the time period before the adversary identifies the source or sink to the length of interval T_S.

- *Latency:* The latency is the average time for a message to travel from source to sink. For simplicity, this is measured using the average hop count of a message from source to sink.

- *Energy consumption:* As a network consumes much more energy on communications than that on computations, this chapter only considers the communication cost. It is assumed that each transmission requires an equal amount of energy and the energy consumption is measured in terms of the average number of messages transmitted in the whole network during the period T_S.

6.3 Location Privacy Protection Schemes Against Local Eavesdropper

The location privacy is vulnerable when the messages travel from a source to sink since the adversary can trace the source or sink by monitoring the packet flow in the network. Thus, the primary purpose for the routing protocols in the delay tolerant event collection system is to protect the location privacy of both the source and sink during the report period T_r. This section describes the proposed four routing schemes for protecting the end-to-end location privacy: forward random walk, bidirectional tree, dynamic bidirectional tree and zigzag bidirectional tree schemes. Each scheme is designed to effectively balance the tradeoff between the safety period, latency and energy consumption.

6.3.1 Forward Random Walk Scheme

In the sample network in Figure 6.1, the source periodically sends messages to the sink by multi-hop wireless communications during the report period T_r. If the messages always travel from the source to the sink along a fixed route, it will be easy for an adversary to capture either the source or the sink via hop-by-hop tracing. Therefore, a solution to achieve the end-to-end location privacy is to randomize the routing path, based on which the *forward random walk* scheme (FRW) is proposed.

The FRW requires all the nodes in the network to obtain their hop counts to the sink, which can be achieved by using a sink-based flooding. At the end of the flooding, each node can get both its own and its neighbors' hop counts to the sink. Let the hop count of node v_i be H_i, then it satisfies $|H_i - H_j| \leq 1$, where node v_j is a neighbor of node v_i and H_j is the hop count of node v_j.

Algorithm 6 Forward Random Walk Scheme (node v_i)

```
Initiation: $Next\_hop = null$.
Build the forward list.
while receive a message do
    Randomly select a neighbor from the forward list as $Next\_hop$.
    Forward the received message to $Next\_hop$.
end while
```

In the FRW scheme, every node divides its neighbors into three lists: *further list*, *equivalent list* and *closer list*. Each neighbor in the further list has a larger hop count than the sender, while each neighbor in the closer list has a smaller hop count than itself. The node' equivalent list consists of neighbors that have the same hop count with itself. The combination of the equivalent list and closer list forms the *forward list*. When forwarding a message, the node will randomly select a neighbor from its forward list as the next hop. Neighbors in the further list will not be considered as the candidates for the next hop since it will remarkably increase the latency. Consequently, the message will be randomly forwarded from the source to sink. Algorithm 6 illustrates the procedure of the FRW scheme.

A forward random route is employed in the FRW scheme. As it is hard for the adversary to identify the actual route to capture the source or sink. Consider that a message is currently held by node v_i, which is d hops away from the sink. Assume the expected number of hops for this message to travel to the sink is x_d, which can be calculated by the following equation:

$$x_d = 1 + x_{d-1}\lambda_d + x_d(1 - \lambda_d) \tag{6.3.1}$$

where λ_d indicates the probability that the message is forwarded from node v_i to a node in its closer list, i.e., the ratio of the size of node v_i's closer list to that of its forward list. From Eq. (6.3.1), it can be further obtained that:

$$x_d = x_{d-1} + \frac{1}{\lambda_d} \tag{6.3.2}$$

When the source is k hops away from the sink, the expected length of the routing path is:

$$\begin{aligned} x_k &= x_{k-1} + \frac{1}{\lambda_k} = x_{k-2} + \frac{1}{\lambda_{k-1}} + \frac{1}{\lambda_k} \\ &= x_0 + \frac{1}{\lambda_1} + \cdots + \frac{1}{\lambda_{k-1}} + \frac{1}{\lambda_k} \end{aligned}$$

It is obvious that $x_0 = 0$, thus,

$$x_k = \sum_{i=1}^{k} \frac{1}{\lambda_i} \tag{6.3.3}$$

Thus, the latency for the FRW scheme is $\sum_{i=1}^{k} \frac{1}{\lambda_i}$ and the energy consumption is also $\sum_{i=1}^{k} \frac{1}{\lambda_i}$.

The FRW scheme protects the location privacy of both the source and sink by randomizing the routing path. However, its latency will be large since the forward random walk lengthens the routing path. Furthermore, the FRW scheme relays messages only to the neighbors in the forward list, resulting in that the safety period can not be very large. A method to achieve a high safety period is to inject *dummy messages* into the network. The *real messages* are defined as the report messages transmitted from the source to sink and the *dummy messages* are the messages with no useful content and they are just generated to draw the adversary away from the actual path.

6.3.2 Bidirectional Tree Scheme

In the hostile network, as the adversary can threat the location privacy of the source or sink by monitoring the packet flow, an effective idea to defend against the threat is to let the source and sink hide in the branches of a tree topology, which requires the adversary to consume more time on discovering them. Therefore, the tree topology in the BT scheme is employed to protect the end-to-end location privacy. Figure 6.3 shows the main idea of the BT scheme. The real messages travel along the shortest path from the source to the sink. To protect the source's location privacy, branches are designed along the shortest path in the source side, in which the dummy messages travel from the leaf nodes to the stalk nodes. The adversary would trace the source by moving backward the direction of the messages, making itself deviate from the real path, which can protect the source's location privacy. Similarly, the branches along the shortest path in the sink side are designed to protect the sink's location privacy. And the dummy messages in the branches travel from the stalk nodes to the leaf nodes, which can draw the adversary away from the real path to protect the sink's location privacy since the adversary would trace the sink by moving forward the direction of the messages.

Algorithm 7 Bidirectional Tree Scheme (node v_i)

```
Initiation: $Next\_hop = Null$, $Child\_node = Null$.
Build the closer list and randomly select a node from the closer list as $Next\_hop$.
$Child\_node \longleftarrow RandomSelect(N_i)$
while receive a real message do
    if $H_i > (1 - \frac{\alpha}{2})H_s$ then
        Send a branch_req message to Child_node with probability of $P$ and TTL=L.
    else if $H_i < \frac{\alpha}{2}H_s$ then
        Send a sink_dummy message to Child_node with probability of $P$ and TTL=L.
    end if
    Forward the message to $Next\_hop$.
end while
```

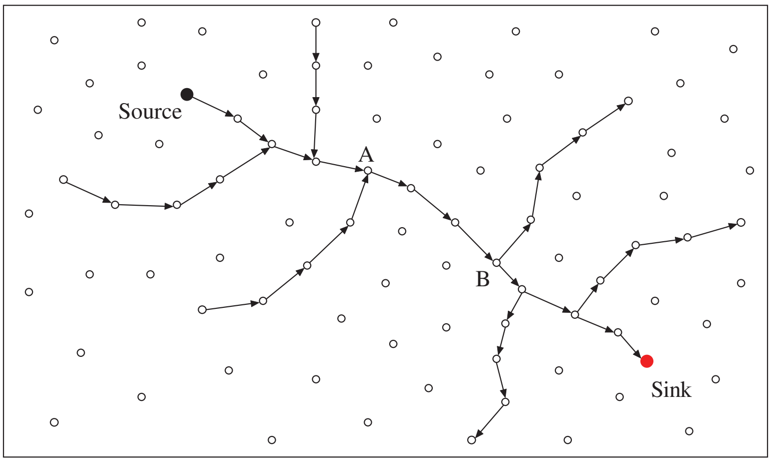


Figure 6.3. The scenario for the bidirectional tree scheme.

Initially, the sink originates a flooding such that each node can obtain the hop count to the sink. Before sending report messages to the sink, the source generates a routing request message including its hop count H_s and sends it to the sink along the shortest path. For each node who receives the routing request message, if its hop count to the sink is larger than $(1 - \frac{\alpha}{2})H_s$, it will randomly select a neighbor with probability of P to generate a branch, where α is the percentage of the nodes on the shortest path to generate the tree branches to protect the location privacy of the source or sink. Meanwhile, for each node who receives the routing request message and its hop count is less than $\frac{\alpha}{2}H_s$, it will randomly select a neighbor with probability of P to generate a branch, which can protect the sink's location privacy.

For instance, if $\alpha = 2/3$, each node on the shortest path with a hop count larger than $\frac{2}{3}H_s$ will originate a branch with probability of P to protect the source's location privacy. And each node on the shortest path with a hop count less than $\frac{1}{3}H_s$ will also originate a branch with the probability P to protect the sink's location privacy. While the nodes in-between just relay the routing request message along the shortest path to the sink.

Algorithm 7 shows the procedure of the BT scheme. When node v_i receives a real message from a neighbor, it relays the message to a neighbor in its *closer list*. In addition, if $H_i > (1 - \frac{\alpha}{2})H_s$, node v_i generates a source side's branch with the probability of P and TTL[6.1] value of L, which indicates the length of branch. Otherwise, if $H_i < \frac{\alpha}{2}H_s$, node v_i generates a sink side's branch with P and L. The generation of the source side's branches and the sink side's branches is described in Algorithm 8 and Algorithm 9 respectively.

Algorithm 8 Source Side Branch Generation (node v_i)

```
Initiation: $Child\_node = Null$, $Parent\_node = Null$.
while receive a $branch\_req$ message do
    Set $Parent\_node$ to the sender of the message.
    if TTL > 0 and $Child\_node = Null$ then
        $Child\_node \longleftarrow RandomSelect(N_i)$
        Forward the $branch\_req$ message to $Child\_node$ with TTL=TTL-1.
    else if TTL = 0 then
        Become a fake source and periodically send $source\_dummy$ message to $Parent\_node$
        with TTL=L.
    end if
end while
while receive a $source\_dummy$ message do
    if TTL > 0 then
        Forward the $source\_dummy$ message to $Parent\_node$ with TTL=TTL-1.
    end if
end while
```

Algorithm 9 Sink Side Branch Generation (node v_i)

```
Initiation: $Child\_node = Null$.
while receive a $sink\_dummy$ message do
    if $TTL > 0$ then
        if $Child\_node = Null$ then
            $Child\_node \longleftarrow RandomSelect(N_i)$
        end if
        Forward the $sink\_dummy$ message to $Child\_node$ with TTL=TTL-1.
    end if
end while
```

The dummy messages in the branches can entice the adversary to get away from the real

[6.1] The TTL used in this chapter is measured in the scale of hop count.

path. Thus, the BT scheme can obtain a long safety period against the local eavesdropper. The parameters α, P and L can be adjusted to get a satisfied performance. If the source is k hops away from the sink, as the real messages travel along the shortest path, the latency is k, indicating that the BT scheme can provide a real time report to the sink. For the energy consumption, in T_S, the average number of transmitted real messages in the network is k, and the average number of transmitted dummy messages is αkPL. So the total energy consumption within T_S is $k \cdot (1 + \alpha PL)$.

Although the BT scheme can obtain end-to-end location privacy against a local eavesdropper, there is still a potential threat. As shown in Figure 6.3, the adversary may be misled, getting lost in the path between A and the source or in the path between B and the sink. However, a powerful adversary may infer the direction of the target based on its visited path $\mathcal{L}$. If the adversary is searching for the source when it is near to B. As the real messages travel along the shortest path, the adversary can trace hop-by-hop from B to A. Then the adversary can infer that the source should be on the extending line of $\overline{BA}$. Thus, the adversary can move directly along $\overline{BA}$ from A and it can identify the source as soon as it gets close enough. The adversary can also use the same strategy to infer the direction of the sink.

6.3.3 Dynamic Bidirectional Tree Scheme

To prevent the adversary from inferring the direction of the source or sink as mentioned above, the dynamic bidirectional tree (DBT) scheme combines the FRW scheme and the BT scheme. Figure 6.4 shows the main idea of the DBT scheme. The paths for the real message vary over time, which greatly increases the tracing difficulty for the adversary as it prevents the adversary from inferring the direction of the target.

Initially, the sink triggers a flooding such that each node can get its hop count to the sink. During the report period T_r, in which the source periodically sends reporting messages to the sink by multi-hop wireless communications, each node who receives the reporting message

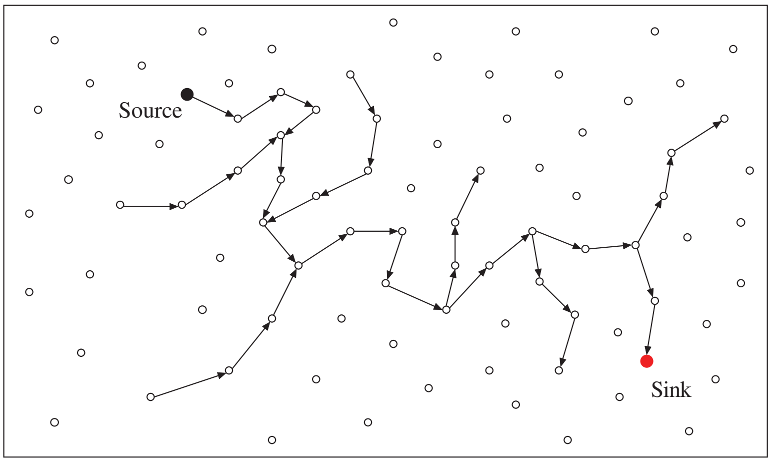


Figure 6.4. The scenario for the dynamic bidirectional tree scheme.

will randomly select a neighbor from its *forward* list to forward the received reporting message. Therefore, the real messages travel with a forward random walk mode from the source to the sink.

To protect the source location privacy, a dynamic tree topology will be adopted. Assume that the hop count of the source is H_s. When a node with the hop count larger than $H_s/2$ receives the real message, it generates a branch with the probability of P. Algorithm 10 shows the procedure of the DBT scheme. When a node v_i receives a real message from its neighbor v_j, it will forward the real message to the next hop, which is randomly selected from its *forward list*. Also, if its hop count is larger than $H_s/2$ but smaller than H_j, it will generate a source side's branch with the probability of P using a method similar to Algorithm 8. The main difference is that each fake source will only send L dummy messages. Otherwise, if its hop count is smaller than both $H_s/2$ and H_j, it will generate a sink side's branch with the probability of P using a method similar to Algorithm 9. The difference is that when a node receives a dummy message, it will reselect a child node to relay this dummy message.

As the real message travels with a forward random walk mode like the FRW scheme, the

Algorithm 10 Dynamic Bidirectional Tree Scheme (node v_i)

```
Initialization: $Next\_hop = null$, $Child\_node = null$.
Build the forward list.
while receive a real message from node $v_j$ do
    Randomly select a node from the forward list as $Next\_hop$ and forward the message to $Next\_hop$.
    $Child\_node \longleftarrow RandomSelect(N_i)$
    if $H_i < H_j$ and $H_i > H_s$ then
        Send a branch_req message to Child_node with probability of $P$ and TTL=L.
    else if $H_i < H_j$ and $H_i < H_s$ then
        Send a sink_dummy message to Child_node with probability of $P$ and TTL=L.
    end if
end while
```

latency for the DBT scheme is $\sum_{i=1}^{k} \frac{1}{\lambda_i}$, where k is the hop count of the source, and λ_i equals to the ratio of the size of node v_i's closer list to that of node v_i's forward list, where node v_i is i hops away from the sink.

6.3.4 Zigzag Bidirectional Tree Scheme

The zigzag bidirectional tree scheme (ZBT) is another location privacy protection scheme, which is proposed to prevent the adversary from inferring the direction of the source or sink. In the ZBT, the proxy source and the proxy sink are employed. The real messages travel along three segments: from the source to the proxy source, from the proxy source to the proxy sink and from the proxy sink to the real sink. As shown in Figure 6.5, concentric circle A represents a proxy source and B represents a proxy sink. In the path from source to A, there will be tree branches to draw the adversary away from the real path. From A to B, the messages will travel along the shortest path. And in the path from B to the sink, there will also be tree branches to protect the sink's location privacy.

To guarantee the effectiveness of the ZBT scheme, two proxy sink candidates are generated, which are deployed at the two opposite sides of the sink. Otherwise, if only one proxy

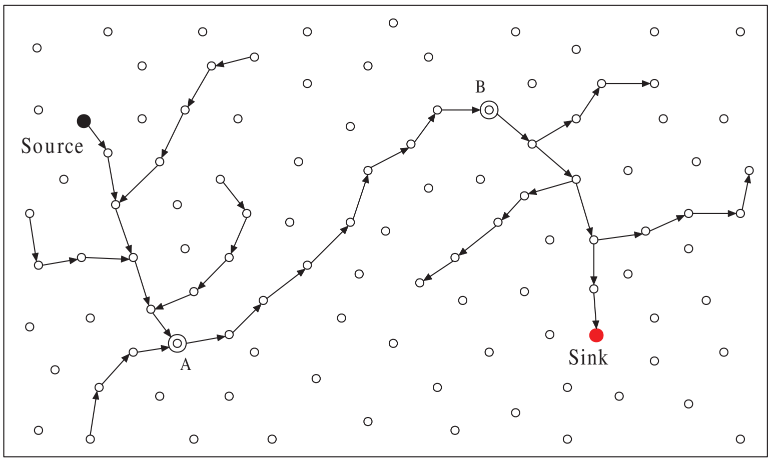


Figure 6.5. The scenario for the zigzag bidirectional tree scheme. Concentric circle A represents a proxy source and B represents a proxy sink.

sink candidate exists and the source is very close to this proxy sink, then the branches on the source side will be very week in protecting the source's location privacy. As shown in Figure 6.6, if the source is close to proxy sink B and proxy sink B is selected as the proxy sink, then the path from the source to proxy source and the path from proxy source to proxy sink B will be very close to each other, it will be vulnerable if the adversary traces from proxy sink B to capture the source as the source is close to proxy sink B. Thus, two candidates can be generated on the two opposite sides of the sink as the proxy sink nodes. The distances between the sink and the proxies can be made approximate to hr, which makes the hop count from each proxy sink to the sink proximate to h. The sink and the two proxy sink nodes initiate flooding for each node to get the hop counts to each of them. As the zigzag routing will be invalid if the proxy sink is close to the source, the source will always select the further candidate as the proxy sink (Proxy sink A is selected in Figure 6.6). To determine the source proxy, the source can initiate a h-hops flooding. Before delivering the report messages to the sink, the source will select a node, which is h hops away from itself, as the source proxy. Note that the proxy source should be carefully selected to make the path from the source to proxy source away from the sink, making the sink safe when the adversary traces along this path.

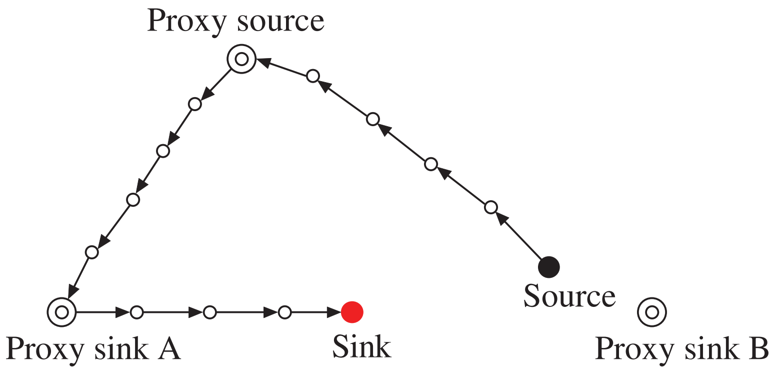


Figure 6.6. Proxy sink and proxy source selection for zigzag bidirectional tree scheme.

Algorithm 11 Zigzag Bidirectional Tree Scheme (node v_i)

```
while receive a real message do
    if destination is proxy source then
        if node $v_i$ is not proxy source then
            Determine the next hop to forward the message towards proxy source, send a
            branch_req message to a node randomly selected from $N_i$ with probability of $P$
            and TTL=L.
        else if node $v_i$ is proxy source then
            Forward the message towards proxy sink.
        end if
    else if destination is proxy sink then
        if node $v_i$ is not proxy sink then
            Forward the message towards proxy sink.
        else if node $v_i$ is proxy sink then
            Forward the message towards sink.
        end if
    else if destination is sink then
        if node $v_i$ is not sink then
            Forward the message towards sink, send a branch_req message to a node randomly
            selected from $N_i$ with probability of $P$ and TTL=L.
        end if
    end if
end while
```

Similar to the BT scheme, as shown in Figure 6.5, when the report message travels between the source and proxy source, each node in the path will generate a branch with probability of

P and TTL of L. When the report message travels from proxy sink to the sink, it will also generate a branch with probability of P and TTL of L. Algorithm 11 describes the procedure of ZBT scheme. Algorithms 8 and 9 describe the source side and the sink side branch generation respectively.

6.4 Performance Evaluation

In this section, the proposed end-to-end location privacy protection schemes for the delay tolerant event collection system will be evaluated on the TOSSIM platform [37] to illustrate their effectiveness.

6.4.1 Performance Metrics and Simulation Settings

The performance evaluation will be conducted with privacy performance metrics, i.e., safety period, latency and energy consumption. The topology of the network is generated by uniformly deploying 3000 sensor nodes within a rectangular area of 30×100. The communication range of each sensor node is 1.67. The average number of neighbor for a node is 8.76. The probability of generating branch of each node who relays the real message is $P = 0.8$, the length of each branch is $L = 10$, α is set as 1 for the BT scheme, and in the ZBT scheme, the hop count of the proxy source to the source is $h = 15$. Two different adversary models, i.e., the patient adversary model and the cautious adversary model are both considered in the simulations. We compare the proposed four location protection schemes with the baseline — the shortest path scheme, in which the message will be forwarded from the source to sink along the shortest path.

6.4.2 Simulation Results

Figure 6.7 shows the safety period of source location privacy under the patient adversary model. It is obvious that the ZBT scheme achieves the highest safety period. The safety period of the

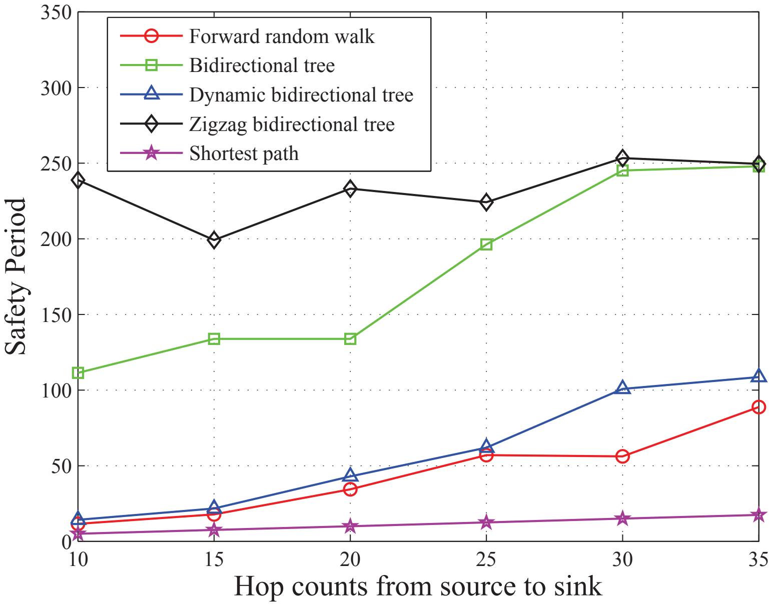


Figure 6.7. Safety period of the source location privacy under patient adversary model.

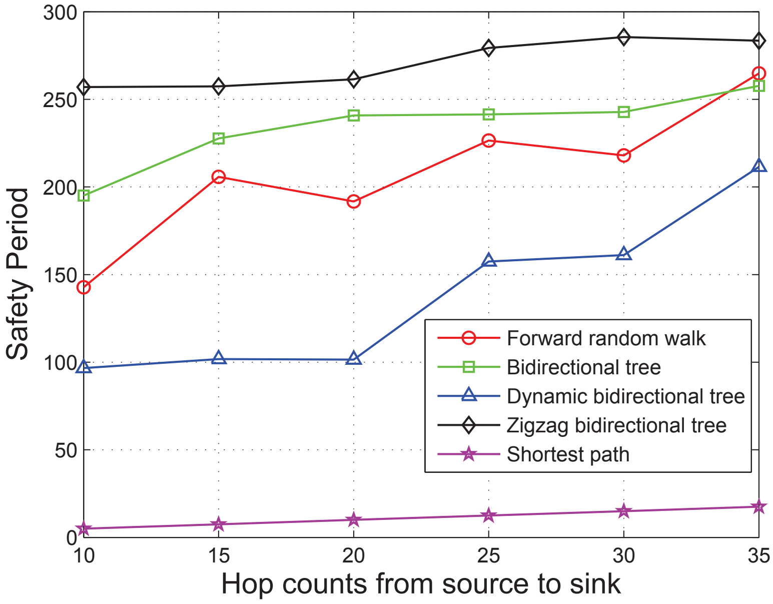


Figure 6.8. Safety period of the sink location privacy under patient adversary model.

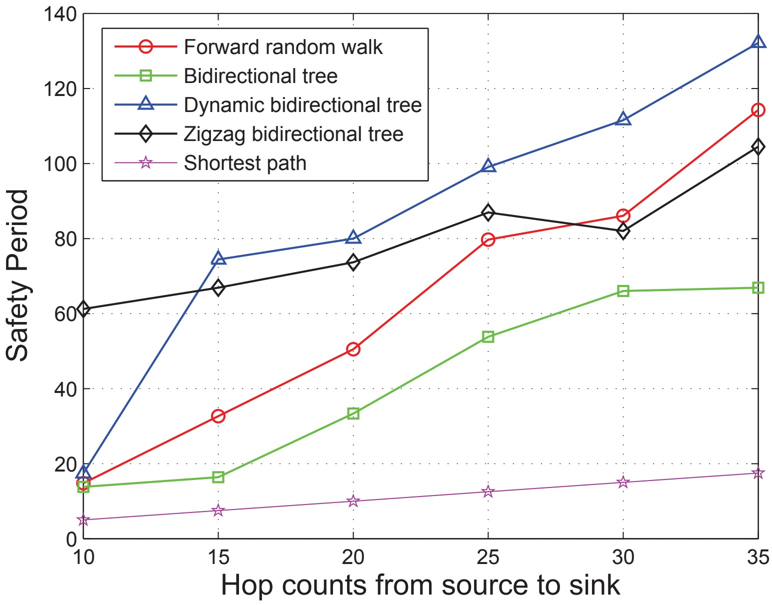


Figure 6.9. Safety period of the source location privacy under cautious adversary model.

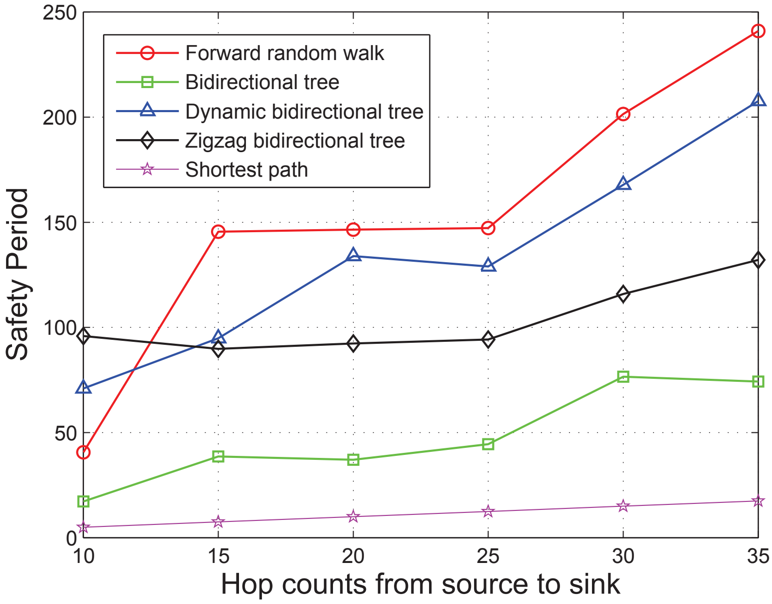


Figure 6.10. Safety period of the sink location privacy under cautious adversary model.

BT scheme increases rapidly as the hop count increases. When the hop count is larger than 30, the safety period of the BT scheme gets close to that of the ZBT scheme. The safety period of the FRW and DBT schemes are relatively low, and the DBT scheme slightly outperforms the FRW scheme. Obviously, the proposed four schemes outperform the shortest path scheme.

In Figure 6.8, the safety period of sink location privacy under the patient adversary model is shown. As the adversary has to determine the direction of the message before it moves to the receiver, which is more time-consuming for the adversary, under the patient adversary model, the safety period of sink location privacy is larger than that of source location privacy. It can be found that the ZBT scheme outperforms other schemes. When the hop count is larger than 15, the safety period of the ZBT, BT and FRW scheme tends to be larger than 200, indicating high sink location privacy. Figure 6.7 and Figure 6.8 illustrate that under the patient adversary model, the DBT scheme can not achieve high location privacy. Also, the proposed four schemes outperform the shortest path scheme. Table 6.1 summarizes the performance comparison of our proposed schemes under the patient adversary model.

Table 6.1. Performance comparison under the patient adversary model.

	Safety period for source location	Safety period for sink location
FRW scheme	relatively low	high
BT scheme	second highest	second highest
DBT scheme	relatively low	relatively low
ZBT scheme	the highest	the highest
Shortest path scheme	the lowest	the lowest

Under the cautious adversary model, the safety period of source location privacy of the proposed schemes and the shortest path scheme is shown in Figure 6.9. The safety period of source location privacy under the cautious adversary is lower than that under the patient adversary model. The reason is that the cautious adversary is smarter and when it waits for a long time at some location, it is able to trace back to avoid being drawn away by some dummy messages. When the hop count equals to 10, the ZBT scheme obtains the highest safety period.

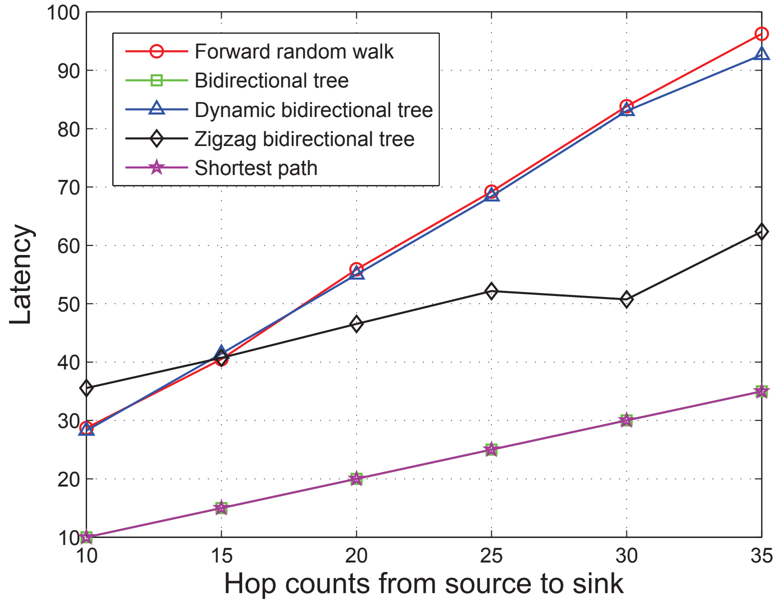


Figure 6.11. Latency of the proposed schemes.

However, the DBT scheme outperforms other schemes when the hop count is larger than 15. The safety period of all the schemes increases with the increase of hop count. However, the increase ratio of the ZBT scheme is the least. Thus, when the hop is larger than 30, the FRW scheme outperforms the ZBT scheme. The performance of the BT scheme is always worse than the FRW, DBT and ZBT schemes. And the proposed four scheme outperform the shortest path scheme.

Table 6.2. Performance comparison under the cautious adversary model.

	Safety period for source location	Safety period for sink location
FRW scheme	relatively low	high
BT scheme	second lowest	second lowest
DBT scheme	the highest	second highest
ZBT scheme	second highest	the highest
Shortest path scheme	the lowest	the lowest

Figure 6.10 illustrates the safety period of sink location privacy under the cautious ad-

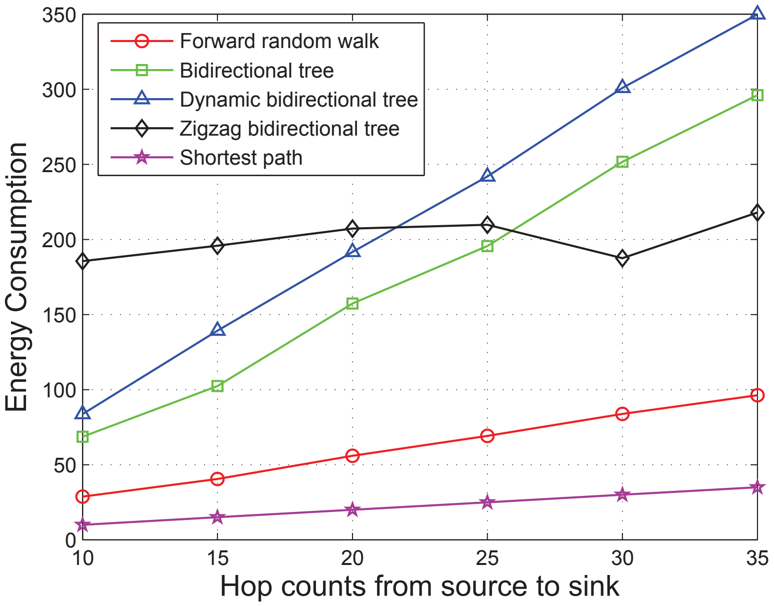


Figure 6.12. Energy consumption of the proposed schemes.

versary model. Compared to the sink location privacy under the patient adversary model, the safety period is lower under the cautious adversary model as the adversary has higher capacity. The FRW scheme achieves the highest performance while the shortest path scheme has the lowest safety period. As it is more time-consuming for the adversary to capture the sink than the source, when under the cautious adversary model, the safety period of sink location privacy is also larger than that of source location privacy. Table 6.2 summarizes the performance comparison of our proposed schemes with the baseline — the shortest path scheme under the cautious adversary model.

Figure 6.11 shows the latency of the four proposed schemes and the shortest path scheme. In the BT scheme and the shortest path scheme, as the real message travels along the shortest path from the source to the sink, it would achieve the shortest latency, which can be validated in Figure 6.11. Since the real message in the FRW and DBT schemes travels along the forward routing path, the latency of these two schemes is similar. When the hop count equals to 10, the latency of the ZBT scheme is the largest as it employs the zigzag path. When the hop count is larger than 15, the latency of the FRW and DBT exceeds that of the ZBT scheme.

Figure 6.12 shows the energy consumption of the proposed schemes and the shortest path scheme. The shortest path scheme consumes the least energy since it does not generate any dummy message and the real message travels along the shortest path. The FRW scheme consumes the second least energy as it also does not generate any dummy message. When the hop count is less than 20, the ZBT scheme consumes the most energy because a farther proxy sink would be selected and more dummy messages are generated than the other schemes. When the hop count is larger than 20, the energy consumption of the DBT scheme is the largest, as more branches are employed than other schemes.

6.5 Summary

The end-to-end location privacy is a significant issue in the delay tolerant event collection systems, which are one of the typical DTN applications. This chapter first addresses the importance of simultaneously protecting the location privacy of both the source and sink in the delay tolerant event collection systems. Then the metrics are defined to evaluate the performance of the end-to-end location privacy protection schemes. Four location privacy protection schemes — forward random walk scheme, bidirectional tree scheme, dynamic bidirectional tree scheme and zigzag bidirectional tree scheme against a local eavesdropper are proposed to obtain the end-to-end location privacy in the delay tolerant event collection systems. The proposed location privacy protection schemes are implemented on the TOSSIM platform, and the performance on the safety period, latency and energy consumption is evaluated. The simulation results illustrate that the proposed location privacy protection schemes can obtain satisfied performance.

CHAPTER 7
SUMMARIZATION

7.1 Conclusions

In this dissertation, the routing problem in the DTNs is studied, based on which the contact expectation is used in designing the DTN routing protocols. An expected encounter based routing protocol (EER), which distributes multiple replicas of a message proportionally between two encounters according to their expected encounter values, is first proposed. In case of single replica of a message, EER makes the routing decision by comparing the minimum expected meeting delay to the destination. To take advantages of the community property, a community based routing protocol (CR), which is divided into inter-community routing and intra-community routing, is further proposed. The EER and CR protocols do not consider the buffer management, thus the default FIFO (first-in-first-out) scheme is adopted. The proposed routing protocols are evaluated in the ONE simulator under different parameters to demonstrate their effectiveness.

This dissertation further considers the group aware DTNs under resource constraints of the bandwidth and buffer space, and proposes a group feature based cooperative routing protocol, which includes the cooperative message transfer scheme and buffer management strategy. In the cooperative message transfer scheme, the constraint of bandwidth is considered and the message transfer priorities are designed to maximize the delivery probability. In the buffer management strategy, by considering the constraint of buffer space, the cooperative message caching scheme is proposed and the dropping order of the messages is designed to minimize the reduced delivery probability. Such kind of buffer management strategy is applicable only in this scenario that two encountering nodes are willing to cooperate with each other. The simulations in the ONE simulator are conducted to illustrate the effectiveness of the proposed cooperative routing protocol.

The selfish characteristic of the nodes in the non-cooperative DTNs is then considered, which require efficient incentive mechanisms to motivate the cooperation of the nodes on the message forwarding. By considering the drawbacks of the previous incentive schemes, the earliest path singular rewarding (EPSR) scheme and earliest path cumulative rewarding (EPCR) scheme are proposed respectively to motivate the nodes to cooperate with each other in the message forwarding. It is proved that the proposed rewarding schemes are incentive compatible, which guarantees that the dominant strategy for the nodes is to truthfully forward the messages. It is also proved that the payment for each delivered message is upper bounded, making the proposed rewarding schemes applicable for the scenario when the nodes have a finite budget. Furthermore, the proposed rewarding schemes are resistant to the malicious behaviors of the selfish nodes. The real trace based simulations are conducted to illustrate the effectiveness of the proposed rewarding schemes.

Finally, this dissertation studies the location privacy problem in the delay tolerant event collection systems, which are one of the typical DTN applications. The importance of simultaneously protecting the location privacy of both the source and sink is addressed. Then the metrics to evaluate the performance of the end-to-end location privacy protection schemes are defined. After that, four location privacy protection schemes — forward random walk, bidirectional tree, dynamic bidirectional tree and zigzag bidirectional tree against a local eavesdropper are proposed to obtain the end-to-end location privacy. The proposed location privacy protection schemes are implemented on the TOSSIM platform, and the performance evaluation based on the safety period, latency and energy consumption is conducted. The simulation results illustrate that the proposed location privacy protection schemes can obtain satisfied performance.

7.2 Future Works

- This dissertation does not consider the construction of the community or group, one of the future directions on the DTN routing protocols will focus on designing the distributed community or group detection method, which is more suitable for the online routing procedure. This dissertation will also intend to design adaptive routing protocols, in which the network parameters such as the number of replicas can be tuned automatically to improve the performance.

- The proposed cooperation incentive mechanisms in this dissertation require the last intermediate node of each delivery path to submit the report message to the TTP, which may introduce a high communication overhead. It will be considered to to reduce this communication overhead in the future work.

- For the location privacy protection schemes, since each of the proposed schemes obtains different performance on protecting the source location privacy or sink location privacy, as the future work, this dissertation plans to decompose the proposed schemes and deeply analyze the effects of each one on the source location privacy and sink location privacy respectively. Then an optimal combination from these decomposed schemes to achieve a highest location privacy protection for both ends will be designed.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A Survey on Sensor Networks. *IEEE Communications Magazine*, 2002.

[2] A. Balasubramanian, B. N. Levine, and A. Venkataramani. DTN Routing As A Resource Allocation Problem. In *Proc. of ACM SIGCOMM*, 2007.

[3] S. Buchegger and J. L. Boudec. Performance Analysis of The CONFIDANT Protocol: Cooperation of Nodes-fairness in Distributed Ad-hoc Networks. In *Proc. of ACM MobiHoc*, 2002.

[4] E. Bulut, S. C. Geyik, and B. K. Szymanski. Conditional Shortest Path Routing in Delay Tolerant Networks. In *Proc. of ACM WoWMoM*, 2010.

[5] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networking. In *Proc. of IEEE INFOCOM*, 2006.

[6] L. Buttyan and J. Hubaux. Stimulating Cooperation in Self-organizing Mobile Ad Hoc Networks. *Mobile Networks and Application*, 2002.

[7] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Impact of Human Mobility on The Design of Oppurtunistic Forwarding Algorithms. In *Proc. of IEEE INFOCOM*, 2007.

[8] B. B. Chen and M. C. Chan. MobiCent: a Credit-Based Incentive System for Disruption Tolerant Network. In *Proc. of IEEE INFOCOM*, 2010.

[9] T. Chen and S. Zhong. INPAC: An Enforceable Incentive Scheme for Wireless Networks using Network Coding. In *Proc. of IEEE INFOCOM*, 2010.

[10] T. Chen, L. Zhu, F. Wu, and S. Zhong. Stimulating Cooperation in Vehicular Ad Hoc Networks: A Coalitional Game Theoretical Approach. *IEEE Transactions on Vehicular Technology*, 2011.

[11] A. Clauset. Finding Local Community Structure in Networks. *Physical Review E*, 2005.

[12] E. Daly and M. Haahr. Social Network Analysis for Routing in Disconnected Delay-Tolerant MANETs. In *Proc. of ACM MobiHoc*, 2007.

[13] J. Deng, R. Han, and S. Mishra. Intrusion Tolerance and Anti-Traffic Analysis Strategies for Wireless Sensor Netowrks. In *Proc. of IEEE DSN*, 2004.

[14] J. Deng, R. Han, and S. Mishra. Countermeasures Against Traffic Analysis Attacks in Wireless Sensor Networks. In *Proc. of ICST SecureComm*, 2005.

[15] C.-H. N. Edith. On Providing Sink Anonymity for Sensor Networks. In *Proc. of ACM International Conference on Wireless Communications and Mobile Computing*, 2009.

[16] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot. Delegation Forwarding. In *Proc. of ACM MobiHoc*, 2008.

[17] L. Eschenaur and V. Gligor. A Key-Management Scheme for Distributed Sensor Networks. In *Proc. of ACM CCS*, 2002.

[18] K. Fall. A Delay-Tolerant Network Architecture for Challenged Internets. In *Proc. of ACM SIGCOMM*, 2003.

[19] K. Fall and S. Farrell. DTN: An Architectural Retrospective. *IEEE Journal on Selected Areas in Communications*, pages 828–836, 2008.

[20] L. C. Freeman. A Set of Measures of Centrality Based on Betweenness. *Sociometry*, pages 35–41, 1977.

[21] L. C. Freeman. Centrality in Social Networks Conceptual Clarification. *Social networks*, pages 215–239, 1979.

[22] W. Gao and G. Cao. On Exploiting Transient Contact Patterns for Data Forwarding in Delay Tolerant Networks. In *Proc. of IEEE ICNP*, 2010.

[23] W. Gao, Q. Li, B. Zhao, and G. Cao. Multicasting in Delay Tolerant Networks: A Social Network Perspective. In *Proc. of ACM MobiHoc*, 2009.

[24] L. He, J. Pan, and J. Xu. An On-Demand Data Collection Scheme for Wireless Sensor Networks with Mobile Elements. In *Proc. of IEEE ICC*, 2011.

[25] Q. He, D. Wu, and P. Khosla. SORI: A Secure and Objective Reputationbased Incentive Scheme for Ad Hoc Networks. In *Proc. of IEEE WCNC*, 2004.

[26] P. Hui, J. Crowcroft, and E. Yoneki. BUBBLE Rap: Social-based Forwarding in Delay Tolerant Networks. In *Proc. of ACM MobiHoc*, 2008.

[27] S. Jain, K. Fall, and R. Patra. Routing in a Delay Tolerant Network. In *Proc. of ACM SIGCOMM*, 2004.

[28] Y. Jian, S. Chen, Z. Zhang, and L. Zhang. Protecting Receiver-Location Privacy in Wireless Sensor Networks. In *Proc. of IEEE INFOCOM*, 2007.

[29] D. Johnson and D. Maltz. Dynamic Source Routing in Ad-hoc Wireless Networks. *Mobile Computing*, pages 152–181, 1996.

[30] E. P. C. Jones, L. Li, and P. A. S. Ward. Practical Routing in Delay-Tolerant Networks. In *Proc. of ACM WTDN*, 2005.

[31] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk. Enhancing Source-Location Privacy in Sensor Network Routing. In *Proc. of IEEE ICDCS*, 2005.

[32] C. Karlof and D. Wagner. Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures. *Elsevier's Ad Hoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, 1(2-3):293–315, 2003.

[33] A. Keranen, J. Ott, and T. Karkkainen. The ONE Simulator for DTN Protocol Evaluation. In *Proc. of the 2nd International Conference on Simulation Tools and Techniques (SIMUTools)*, 2009.

[34] I. N. Kovalenko and A. I. Kochubinskii. Asymmetric Cryptographic Algorithms. *Cybernetics and Systems Analysis*, 2003.

[35] A. Krifa, C. Barakat, and T. Spyropoulos. An Optimal Joint Scheduling and Drop Policy for Delay Tolerant Networks. In *Proc. of ACM WoWMom*, 2008.

[36] A. Krifa, C. Barakat, and T. Spyropoulos. Optimal Buffer Management Policies for Delay Tolerant Networks. In *Proc. of IEEE SECON*, 2008.

[37] P. Levis, N. Lee, M. Welsh, and D. Culler. Tossim: Accurate and Scalable Simulation of Entire Tinyos Applications. In *Proc. of ACM SenSys*, 2003.

[38] F. Li and J. Wu. FRAME: An Innovative Incentive Scheme in Vehicular Networks. In *Proc. of IEEE ICC*, 2009.

[39] Q. Li, S. Zhu, and G. Cao. Routing in Socially Selfish Delay Tolerant Networks. In *Proc. of IEEE INFOCOM*, 2010.

[40] X. Li, W. Shu, M. Li, H. Huang, and M. Y. Wu. DTN Routing in Vehicular Sensor Networks. In *Proc. of IEEE GLOBECOM*, 2008.

[41] X. Y. Li, Y. Wu, P. Xu, G. Chen, and M. Li. Hidden Information and Actions in Multi-Hop Wireless Ad Hoc Networks. In *Proc. of ACM MobiHoc*, 2008.

[42] Y. Li, M. Qian, D. Jin, L. Su, and L. Zeng. Adaptive Optimal Buffer Management Policies for Realistic DTN. In *Proc. of IEEE GLOBECOM*, 2009.

[43] Y. Li and J. Ren. Preserving Source-Location Privacy in Wireless Sensor Networks. In *Proc. of IEEE SECON*, 2009.

[44] Y. Li and J. Ren. Source-Location Privacy through Dynamic Routing in Wireless Sensor Networks. In *Proc. of IEEE INFOCOM*, 2010.

[45] D. Liben-Nowell and J. Kleinberg. The Link pPrediction Problem for Social Networks. In *Proc. of ACM CIKM*, 2003.

[46] A. Lindgren, A. Doria, and O. Schelen. Probabilistic Routing in Intermittently Connected Networks. In *Proc. of ACM MobiHoc*, 2003.

[47] A. Lindgren and K. S. Phanse. Evaluation of Queuing Policies and Forwarding Strategies for Routing in Intermittently Connected Networks. In *Proc. of IEEE COMSWARE*, 2006.

[48] C. Liu and J. Wu. Scalable Routing in Delay Tolerant Networks. In *Proc. of ACM MobiHoc*, 2007.

[49] C. Liu and J. Wu. Routing in a Cyclic MobiSpace. In *Proc. of ACM MobiHoc*, 2008.

[50] C. Liu and J. Wu. An Optimal Probabilistic Forwarding Protocol in Delay Tolerant Networks. In *Proc. of ACM MobiHoc*, 2009.

[51] M. Lukac, V. Naik, I. Stubailo, A. Husker, and D. Estrin. In Vivo Characterization of a Wide Area 802.11b Wireless Seismic Array. *Center for Embedded Networked Sensing*, 2007.

[52] P. S. M. Reed and D. Goldschlag. Anonymous connections and onion routing. *IEEE Journal on Selected Areas in Communication. Special issue on copyright and privacy protection*, 1998.

[53] Y. Ma and A. Jamalipour. A Cooperative Cache-Based Content Delivery Framework for Intermittently Connected Mobile Ad Hoc Networks. *IEEE Transactions on Wireless Communications*, pages 366–373, 2010.

[54] M. E. Mahmoud and X. Shen. PIS: A Practical Incentive System for Multi-hop Wireless Networks. *IEEE Transactions on Vehicular Technology*, pages 4012–4025, 2010.

[55] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. Wireless Sensor Networks for Habitat Monitoring. In *Proc. of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, 2002.

[56] K. Mehta, D. Liu, and M. Wright. Location Privacy in Sensor Networks Against a Global Eavesdropper. In *Proc. of IEEE ICNP*, 2007.

[57] S. C. Nelson, M. Bakht, and R. Kravets. Encounter-Based Routing in DTNs. In *Proc. of IEEE INFOCOM*, 2009.

[58] S. C. Nelson, A. F. Harris, and R. Kravets. Event-driven, Role-based Mobility in Disaster Recovery Networks. In *Proc. of ACM CHANTS*, 2007.

[59] M. E. J. Newman. Analysis of Weighted Networks. *Physical Review E*, 2004.

[60] M. E. J. Newman. Analysis of Weighted Networks. *Physical Review E*, 2004.

[61] M. E. J. Newman. Detecting Community Structure in Networks. *The European Physical Journal B - Condensed Matter and Complex System*, 38(2):321–330, March 2004.

[62] M. E. J. Newman and M. Girvan. Finding and Evaluating Community Structure in Networks. *Physical Review E*, 2004.

[63] Y. Ouyang, Z. Le, G. Chen, J. Ford, and F. Makedon. Entrapping Adversaries for Source Protection in Sensor Networks. In *Proc. of ACM WoWMoM*, 2006.

[64] Y. Ouyang, Z. Le, D. Liu, J. Ford, and F. Makedon. Source Location Privacy against Laptop-Class Attacks in Sensor Networks. In *Proc. of ICST SecureComm*, 2008.

[65] C. Ozturk, Y. Zhang, and W. Trappe. Source-Location Privay in Energy Constrained Sensor Network Routing. In *Proc. of ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN)*, 2004.

[66] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, pages 814–818, 2005.

[67] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering The Overlapping Community Structure of Complex Networks in Nature and Society. *Nature*, 2005.

[68] G. Pei, M. Gerla, and X. Hong. LANMAR: Landmark Routing for Large Scale Wireless Ad Hoc Netowrks with Group Mobility. In *Proc. of ACM MobiHoc*, 2000.

[69] L. Pelusi, A. Passarella, and M. Conti. Opportunistic Networking: Data Forwarding in Disconnected Mobile Ad Hoc Networks. *IEEE Communication Magazine*, pages 134–141, 2006.

[70] C. E. .Perkins and P. Bhagwat. Higly Dynamic Destination-sequenced Distance-vector Routing (DSDV) for Mobile Computers. In *Proc. of ACM SIGCOMM*, 1994.

[71] C. E. Perkins and E. M. Royer. Ad-hoc On-demand Distance Vector Routing. In *Proc. of IEEE WMSCA*, 1999.

[72] A. Pfitzmann and M. Kohntopp. Anonymity, unobservability and pseudonymity - a proposal for terminology. In *Proc. of International Workshop on the Design Issues in Anonymity and Observability*, 2000.

[73] A. K. Pietilainen and C. Diot. Social Pocket Switched Networks. In *Proc. of IEEE INFOCOM Workshops*, 2009.

[74] J. M. Pujol, A. L. Toledo, and P. Rodriguez. Fair Routing in Delay Tolerant Networks. In *Proc. of IEEE INFOCOM*, 2009.

[75] N. M. R. Dingledine and P. Syverson. Tor: The second generation onion router. In *Proc. of 13th USENIX Security Symposium*, 2000.

[76] M. Shao, Y. Yang, S. Zhu, and G. Cao. Towards Statistically Strong Source Anonymity for Sensor Networks. In *Proc. of IEEE INFOCOM*, 2008.

[77] L. Song, D. Kotz, R. Jain, and X. He. Evaluation Location Predictors with Extensive Wi-Fi Mobility Data. In *Proc. of IEEE INFOCOM*, 2004.

[78] T. Spyropoulos, K. Psounis, and C. Raghavendra. Spray and Wait: An Efficient Routing Scheme for Intermitently Connected Mobile Networks. In *Proc. of ACM WDTN*, 2005.

[79] T. Spyropoulos, K. Psounis, and C. Raghavendra. Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility. In *Proc. of PerCom Workshops*, 2007.

[80] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Single-copy Routing in Intermittently Connected Mobile Networks. In *Proc. of IEEE SECON*, 2004.

[81] S. Srinivasa and S. Krishnamurthy. CREST: An Opportunistic Forwarding Protocol Based on Conditional Residual Time. In *Proc. of IEEE INFOCOM*, 2009.

[82] T. E. O. Systems. Center for Embedded Networked Sensing, UCLA. In *http://research.cens.ucla.edu/areas.2007/Terrestrial*, 2007.

[83] M. Y. S. Uddin, B. Godfrey, and T. Abdelzaher. RELICS: In-network Realization of Incentives to Combat Selfishness in DTNs. In *Proc. of IEEE ICNP*, 2010.

[84] A. Vahdat and D. Becker. Epidemic Routing for Partially Connected Ad Hoc Networks. In *Technical Report CS-200006*, Duke University, April 2000.

[85] H. Wang, B. Sheng, and Q. Li. Privacy-aware Routing in Sensor Networks. *Computer Networks*, 53(9):1512–1529, 2009.

[86] W. Wang, L. Chen, and J. Wang. A Source-location Privacy Protocol in WSN Based on Locational Angle. In *Proc. of IEEE ICC*, 2008.

[87] J. Wu, S. Yang, and F. Dai. Logarithmic Store-carry-forward Routing in Mobile Ad Hoc Networks. *IEEE Transactions on Parallel and Distributed Systems*, 2007.

[88] Y. Xi, L. Schwiebert, and W. Shi. Preserving Source Location Privacy in Monitoring-based Wireless Sensor Networks. In *Proc. of the 2nd International Workshop on Security in Systems and Networks (SSN)*, 2006.

[89] K. Xu, G. H. Yang, V. O. K. Li, and S. Y. Chan. Detecting Dynamic Communities in Opportunistic Networks. In *Proc. of the first international conference on Ubiquitous and future networks*, 2009.

[90] X. Xu, J. Luo, and Q. Zhang. Delay Tolerant Event Collection in Sensor Networks with Mobile Sink. In *Proc. of IEEE INFOCOM*, 2010.

[91] Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao. Towards Event Source Unobservability with Minimum Network Traffic in Sensor Networks. In *Proc. of ACM WiSec*, 2008.

[92] Q. Yuan and J. Wu. Predict and Relay: An Efficient Routing in Disruption-Tolerant Networks. In *Proc. of ACM MobiHoc*, 2009.

[93] X. Zhang, J. F. Kurose, B. Levine, D. Towsley, and H. Zhang. Study of A Bus-Based Disruption Tolerant Network: Mobility Modeling and Impact on Routing. In *Proc. of ACM MobiCom*, 2007.

[94] S. Zhong, L. Li, Y. G. Liu, and Y. R. Yang. On Designing Incentive-Compatible Routing and Forwarding Protocols in Wireless Ad-Hoc Networks—An Integrated Approach Using Game Theoretical and Cryptographic Techniques. In *Proc. of ACM MobiCom*, 2005.

[95] H. Zhu, X. Lin, R. Lu, Y. Fan, and X. Shen. SMART: A Secure Multilayer Credit-Based Incentive Scheme for Delay-Tolerant Networks. *IEEE Transactions on Vehicular Technology*, 2009.