Message Forwarding and Scheduling in Delay Tolerant Networks

by

Ahmed Elwhishi

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2012

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Delay-tolerant networking (DTN) has recently received considerable attention from the research community. This type of networks is characterized by frequent disconnections due to propagation phenomena, node mobility, and power outages. Thus, the complete path between the source and the destination may never have existed. This context requires the design of new communication paradigms and techniques that will make communication possible in these environments. To achieve message delivery, researchers have proposed the use of store-carry-and-forward protocols, whereby a node may store the message and carry it until an appropriate forwarding opportunity arises. Many flooding-routing schemes have been proposed for DTNs in order to increase the probability of message delivery. However, these schemes suffer from excessive energy consumption, severe contention that significantly degrades their performance, especially if we account for the fact that each node could be a hand-held and battery-powered device with stringent buffer size limitation. With such buffer limitations at the DTN nodes, message drop/loss could happen due to buffer overflow.

In order to address the problem and improve the performance of DTNs, this thesis focuses on two main design objectives; first, the design and evaluation of new multi-copy routing schemes; second, the design and evaluation of new scheduling and dropping policies to reduce message drop/loss due to buffer overflow. To fulfill the first objective, a protocol called Self Adaptive Routing Protocol (SARP) is introduced. It is a multi-copy scheme designed to suit resource-sufficient DTNs. Based on SARP, two multi-copy routing schemes are further developed to suit resource-limited DTNs, in which compensating the traffic demand become a challenge: i) the Self Adaptive Utility-based Routing Protocol (SAURP), ii) and the Adaptive Reinforcement based Routing Protocol (ARBRP). The introduced protocols form a new framework of DTNs aiming to significantly reduce the resource

requirements of flooding-based routing schemes. Each introduced scheme has its own way of exploring the possibility of taking mobile nodes as message carriers in order to increase the delivery ratio of the messages. In SAURP, the best carrier for a message characterized by jointly considering the inter-contact time that is obtained using a novel contact model and the network status, such as including wireless link condition and nodal buffer availability. In ARBRP, the routing problem is solved by manipulating a collaborative reinforcement learning technique, where a group of nodes can cooperate with each other to make a forwarding decision for the stored messages based on a cost function at each contact with another node. ARBRP is characterized by not only considering the contact time statistics, but also looks into the feedback on user behavior and network conditions, such as congestion and buffer occupancy sampled during each previous contact with any other node. The thesis argues and proves that the nodal movement and the predicted collocation with the message recipient can serve as meaningful information to achieve an intelligent message forwarding decision at each node. Therefore, the introduced protocols can achieve high efficiency via an adaptive and intelligent routing mechanism according to network conditions.

To fulfill the second objective, we further enhanced the performance of DTN routing by introducing message scheduling and dropping policies such that the delivery ratio is increased and/or the delivery delay is reduced. This thesis investigates new buffer management and scheduling policies to improve the performance of flooding and utility-based forwarding routing in DTNs, such that the forwarding/dropping decision can be made at a node during each contact for either optimal message delivery ratio or message delivery delay.

To examine their effectiveness, the introduced protocols and the buffer management and scheduling policies have been implemented and compared to a number of existing counterpart approaches. A near-realistic mobility model is used for testing. A number of scenarios are used to evaluate the performance of the introduced techniques in terms of delivery delay, ratio, and the number of transmissions performed.

Acknowledgments

First and foremost, I would like to express my sincere appreciation to my supervisors, Prof. Pin Han Ho and K. Naik. I thank them for their continuing guidance and support during my whole PhD. research. Without their vision, advices, strong analytical skills and commitment to the excellence, this thesis would not have been possible. Their great enthusiasm has been the main supporting power for the completion of this thesis. I have learned so many things from them not only on the technical side but also on the personal side. I sincerely thank them for encouraging and inspiring me in each step, their encouragement have made this research and our collaboration an exciting experience which I am genuinely proud of.

I would like to express my appreciation to my thesis committee members: Prof. Otman Basir, Prof. Ali Elkamel, Prof. Chi-Hsiang Yeh, and Prof. Catherine Gebotys. They have devoted precious time and efforts for my thesis and have helped me to improve it. Their comments and invaluable suggestions are genuinely appreciated. Also, I thank and express my gratitude to all the staff of the department of Electrical and Computer Engineering, for they are always helpful and resourceful.

Finally, I would like to dedicate this thesis to my beloved wife Fawzia, and my children, Ayub, Ryem, and Adam. I hope I can make it up to them for all the time they sacrificed to help me to be devoted to my study.

Contents

List of Tables

List of Figures

List of Notations

$V(s)$	The cost function
$\triangle T$	The inter-contact time
CS	The contact statistics
W	The time window
T_{total}	The total contact time
T_{busy}	The time in which the channel is busy
T_{free}	The time in which the channel is free
β	The contact time rate
$m_i(t)$	The nodes who have seen message i
$n_i(t)$	The nodes who have copy of message i
T_i	The time elapsed since creation of message i
R_i	The remaining time-to-live of message i
Tx	The time-to-live of the message

List of Abbreviations

DTN	Delay Tolerant Networks
ICMN	Intermittently Connected Mobile Networks
MANET	Mobile Ad hoc Networks
IPN	Inter-Planetary Networks
PSN	Pocket Switched Networks
UAV	Unmanned Aerial Vehicles
MMF	Most Mobile First
MSF	Most Social First
CBMM	Community_based Mobility Model
LSF	Last Seen First
SARP	Self Adaptive Routing Protocol
SAURP	Self Adaptive Utility based Routing Protocol
UCUM	Utility-function Calculation and Update Module
TUM	Transitivity Update Module
FSM	Transitivity Update Module
ARBRP	Adaptive Reinforcement Based Routing Protocol
CRL	Collaborative Reinforcement Learning
RWP	Random Way Point
S&F	Spray and Focus
S&W	Spray and Wait

CSMA	Carrier Since Multiple Access
CDF	Connectivity Degree Feature
SVEM	Summary Vector Exchange Module
NSEM	Network State Estimation Module
UCM	Utility Calculation Module
GKM	Global Knowledge-based Management
EHP	Encounter History-based Prediction
SFDP	SAURP Forwarding and Dropping Policy
GKM	Global Knowledge-based Management
DF	Drop Front
HBD	History Based Drop
DF	Drop Front

Chapter 1

Introduction

1.1 Overview of DTNs

The widespread adaptation and employment of wireless technologies means that a wide range of devices can be interconnected over vast distances through wireless links. As successful as these networks have been, they nonetheless cannot reach everywhere, and for some applications, the high cost of the associated scenarios is prohibitive. One of the most serious challenges arises in cases in which network connectivity cannot be guaranteed. For such challenged networking environments, the current networking technology relies on a set of fundamental assumptions that are not true in all practical environments. The first and most important fundamental assumption is the existence of a direct end-to-end path from a source to a destination. This assumption can easily be violated due to nodal mobility, power-saving policies, or unreliable packet delivery strategies. As a result, the mechanism of the TCP/IP-based network model that provides end-to-end communication is not valid, so any synchronous communication paradigm is likely perform very poorly. For these challenged networking environments, such as those found in mobile in-motion networks and dynamic wireless networks, network connectivity is rather opportunistic.

Techniques for producing applications that can tolerate disruptions and/or high delays in network connectivity are essential for these opportunistic networks. Networks that include such applications are often collectively referred to as Intermittently Connected Mobile Networks (ICMNs) or Delay Tolerant Networks (DTNs) [1, 26]. Many real ICMNs fall into this category, such as Military Networks [3], Inter-Planetary Networks (IPN)[2], Pocket Switched Networks (PSN)[4], wildlife tracking and habitat-monitoring sensor networks [6], and networks that provide low-cost Internet service to remote communities [7, 72, 15]. These networks belong to the general category of DTN, in which delays incurred are unpredictable and can be very long. This situation arises because of sparse network topologies, node heterogeneity, and volatile link conditions that are possibly due to wireless propagation phenomena and node mobility. As a result, network links may be mostly disconnected or highly susceptible to a variety of disruptions that cause them to perform a set of disconnected clusters of nodes. To achieve eventual delivery, some nodes must store messages and wait for the opportunity to forward the interrupted messages.

1.2 Routing Challenges in DTNs

Routing is one of the most fundamental problems in dealing with intermittently connected networks. In contrast to the routing schemes in Mobile Ad-hoc Networks (MANETs) such as Dynamic Source Routing (DSR), Ad hoc On-demand Distance Vector (AODV), or Optimized Link State Routing Protocol (OLSR) [8], a DTN may lack an end-to-end path for a given node pair for most of the time. Protocols developed for MANETs are therefore unable to address the intrinsic characteristics of a DTN. The complete-path discovery mechanism may fail in reactive schemes,

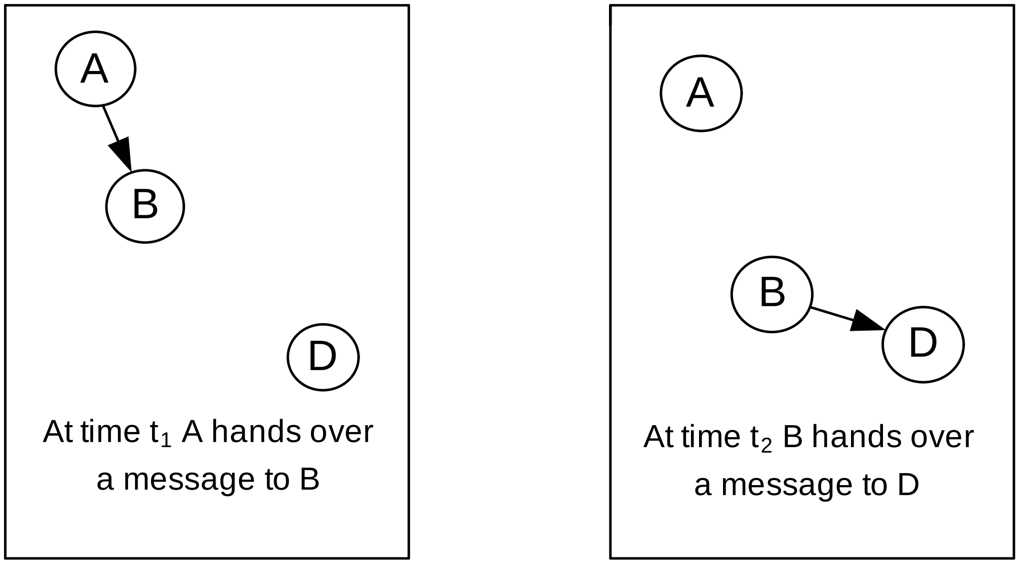


Figure 1.1: Store carry forward mechanism

while convergence of proactive protocols results in a deluge of topology update messages.

To cope with frequent, long-lived disconnections and deal with variation in the links over time (different links may come up and down due to node mobility), a node can buffer the message and wait until it finds an existing link to the next hop that is to store the message and wait for another existing link, and so on, until the message reaches its destination. Figure 1.1 illustrates the principle of the store, carry, and forward mechanism. This type of connectivity imposes a new paradigm for routing mechanisms: mechanisms based on current connectivity information and predictions of future connectivity, which plays an important role in forwarding decisions. In addition, node mobility and network topology need to be exploited so that a message can reach its destination.

What the above two mechanisms share is the exploitation of node mobility to carry messages around the network as part of the routing algorithm. These schemes are collectively referred to as encounter-based routing (in related literature they are also referred to as mobility-assisted or store-carry-and-forward). Encounter-based routing consists of a series of independent forwarding decisions that take place when two nodes meet, and these nodes are completely oblivious of the specific path the message will eventually follow. In this paradigm, nodes carry a set of messages, possibly for long periods of time, until other nodes are encountered; exchange messages according to a specific protocol; and then continue their trip.

Depending on the number of copies of a single message that may coexist in the network, one can define two major categories of encounter-based routing schemes: single-copy and multi-copy. In single-copy schemes, each message has only one node in the network that carries a copy of the message at any given time. This node is called the "custodian" of the message. When the current custodian forwards the copy to an appropriate next hop, the new node becomes the message's new custodian, and so on, until the message reaches its destination. However, the main challenge that faces these schemes is how to be very efficient in dealing with an interruption in network connectivity. On the other hand, multiple-copy (or multi-copy) routing schemes may generate multiple copies of the same message, which can then be routed independently in order to increase robustness and performance. However, they consume a large amount of energy, bandwidth, and memory (buffer) space. In addition, they suffer from contention in case of a high traffic load, when packet drops can significantly degrade performance and scalability.

This thesis presents routing protocols designed for easy deployment. They must therefore meet three design goals. First, the routing must be self-configuring, which is critical for equipment that may be deployed far from network experts and for maintaining communication capability even when some components fail. Both of those problems occur in many application domains in which DTNs can provide significant benefits. Second, the protocol must provide acceptable performance for a wide variety of connectivity patterns, which implies that the protocol is a good choice for most DTN scenarios. This feature eliminates the need for analysis to determine which protocol to use. Finally, the protocol must make efficient use of buffer and network resources. If the DTN becomes a valuable resource, it will be used frequently by a large number of users and must therefore be capable of scaling with the demand.

1.3 Research Objectives

In DTNs, message delivery depends on the traffic patterns, buffer capacity, and scalability provided by the routing protocol. This thesis presents encounter-based routing protocols that attempt to use the information available in order to deliver messages between nodes without the existence of a complete end-to-end path. These protocols are characterized by efficient use of mobility information, bandwidth, and buffer space. The main objectives in this research can be summarized as follows:

- Formulating an architectural framework for DTNs that supports routing in large-scale scenarios in which nodes are in sparsely populated communities and no continuous end-to-end path exists.

- Developing a novel routing protocols that achieves data delivery using the concept of history of previous encounters and contact durations in an attempt to achieve maximum stability in a structure.

- Improving the capacity utilization of DTNs by employing a new updating rule that makes effective use of the utility function employed in existing DTN encounter-based routing.

- Reducing delivery delays by employing an efficient decision-making policy for the forwarding of messages when two nodes come within transmission range of each other.

- Proposing a framework that will operate when the traffic demand becomes higher because the number of copies of each message must be spread throughout the network in order to meet the delivery deadline for some messages and to avoid violating the limited buffer space of the nodes. An attempt will be made to show that such redundancy is necessary in order to achieve the desired performance in the highly problematic context of intermittently connected networks.

- Proposing a framework for buffer management to avoid violating the limited buffer space of the nodes as the traffic demand becomes higher. An attempt will be made to introduce scheduling and drop policy in order to reduce the delivery delay and/or increase delivery ratio of the messages.

1.4 Thesis Outline

This proposal is organized as follows. Chapter 1 introduces DTNs and the basic concept of encounter-based routing protocols as well as the motivation and objectives of this research. Chapter 2 provides the background material for this research. Chapter 3 presents the problem description and solution approach. Chapter 4 describes the introduced SARP that suits a network with light-traffic load, and its application for solving routing problems in DTNs. An analysis of its performance is also presented. Chapter 5 presents the introduced multi-copy routing techniques that their mechanism can accommodate to some degree the high traffic loads and contention along with an analysis of their performance. Chapter 6 presents buffer management and scheduling framework to improve the performance of DTNs routing. Chapter 7 summarizes the thesis, and provides interesting and challenging directions for future research.

Chapter 2

Background and Related Work

Routing is one of the very challenging open issues with respect to DTNs. The routing problem in DTNs [9] has been under extensive study during the last few years. The research includes studies conducted before the term "delay-tolerant" was widely used. The adjectives "intermittently-connected," "disruption-tolerant," "sparse," and "disconnected" were also used to describe problems with networks in scenarios in which partitions are frequent and a connected path between the source and the destination may not be present, such as satellite and interplanetary communication systems. Despite a significant amount of work on consensus with respect to general DTN architecture [1], there has been no similar focus or agreement about DTN routing algorithms, especially when for networks with "opportunistic" connectivity. The first DTN architecture to solve Internet-working issues was proposed by Fall et al. [10], following which the DTN research group [1] standardized DTN architecture by proposing an RFC.

This chapter follows the taxonomy introduced by Jones et al. [11] to present two main categories for classifying delay-tolerant routing: replication based and knowledge based. Replication-based approaches address the ways messages can be disseminated among several relays in order to increase their chance of reaching their destinations. They also describe how a routing strategy relies on multiple copies of each message. Knowledge-based approaches describe how information about the network is used for making decisions. The information a node obtains about the connectivity and behavior of network nodes is used to make efficient forwarding decisions that improve routing performance.

Some protocols in this category employ both controlled replication and knowledge in their forwarding policies. These schemes are also classified based on the number of copies of each message. A survey of routing protocols for intermittently connected DTNs can be found in [9]. Before DTN routing is addressed in greater detail, an overview of the most important properties of DTNs is provided, including the metrics used to evaluate the effectiveness of routing techniques. Each routing approach is then classified based on the two categories mentioned.

2.1 Network Model

For routing problem to be solved, a model of the network is needed so that its behavior and characteristics can be described. A DTN is composed of computing devices, called nodes that participate in the network. The network suffers from frequent disruptions in its connectivity so that the topology is only intermittently and partially connected. The one-way links that connect some nodes together are subject to intermittent connectivity (links go up and down) over time due to power constraints, mobility, failures, or other events. When the link is up, the source node has an opportunity to send data to the other end. In DTN literature, this opportunity is called a contact [10]. A node can have more than one contact with other nodes. The contact schedule is the set of times during which the contact is available. In DTN architecture [1], complete messages are forwarded over each hop. The intermediate nodes buffer the messages if their storage capacity is not violated.

This mechanism enables messages to wait until they find an appropriate next hop, which may take a long time.

As explained by Jain et al. [12], four components are used to calculate the total amount of time for one message to be transmitted from one node to another: waiting time, queuing time, transmission delay, and propagation delay. The waiting time is how long a message must wait at a node until the contact to the next node becomes available, that is, the next node comes into range. This interval depends on the contact schedule and the message arrival time. The queuing time is the time it takes to drain the queue of higher-priority messages, which depends on the amount of competing traffic in the network and the contact data rate. The transmission delay is the time it takes for all the bits of the message to be transmitted, which can be computed from the contact's data rate and the length of the message. The propagation delay is the time it takes a bit to propagate across the connection, which depends on the link technology.

2.1.1 Challenges

Delay-tolerant networks give rise to many challenges not present with traditional networks. These challenges result from the need to deal with disconnections, which have a direct impact on routing and forwarding, and on limited resources.

Network Connectivity

As mentioned in a DTN, a message may be buffered at a node waiting for its next hop to become available. Buffering can range from seconds to days whereas in the other types of networks the delays are typically much shorter. Thus, this study must consider the nature of the underlying network connectivity, which depends strongly on the application area under consideration. Patterns of node contacts can be classified based on how predictable they are:

- Precise contacts: These contacts can exist, for instance, between a base-stations located somewhere on earth and a low-earth orbiting relay satellite, where disconnections are caused by the movement of objects in space, which can be calculated very accurately.

- Approximate scheduling: these contacts are scheduled but with some expected delays caused by varying arrival times. Examples of these DTNs are the nodes mounted on city buses. These buses have a schedule but are subject to delays due to traffic, equipment failure, natural disasters, or accidents. Their actual arrival times thus vary significantly.

- On-demand contacts: This type of connectivity reinforces connectivity on demand by bringing, for example, additional communication resources into a network when necessary, e.g., satellites, Unmanned Aerial Vehicles (UAVs). Similarly, one could force a number of specialized nodes, e.g., robots to follow a given trajectory between disconnected parts of a network in order to bridge a gap [13, 14, 22].

- Predicted contacts: These contacts are not scheduled, but a prediction of their existence can be made by analyzing the past history of node movement or by using a hypothesis about node movement. One example that fall into this category is human activity. It can be argued that recurrence is a common property of mobility models in DTNs. For example, humans tend to perform repetitive tasks, such as going to soccer games, working, grocery shopping, and attending entertainment events. Workers often repeat tasks and activities have commitments to specific clients, meet at specific locations, run specific types of errands, etc. Many mobile agents have a small set of frequently revisited destinations, e.g., cars revisit gas stations, trucks deliver goods to specific locations, birds return to their nests, and animals frequently go to the same water sources. Although they have predictable, fairly regular schedules, there is no guarantee of a specific time when a person will be at work.

- Opportunistic contacts: These contacts exhibit completely random connectivity which is neither enforced nor predictable and which is subject to the statistics of the mobility model. These networks are widely studied in the ad hoc networking community because the models are simple to work with. This classification is similar to the one presented in [16]. The aim of this study is to investigate solutions for situations in which connectivity is neither enforced nor scheduled but is subject to the statistics of node movements. The focus is on predicted contacts since the real mobility of human movement falls into this category and is challenging to study [17, 54]. Nodes in networks with this type of mobility can be represented as contacts that can be predictably brought up or down. The routing technique presented in this thesis is also applicable to networks with predicted mobility.

Contact Capacity

The contact capacity depends on the link technology and the duration of the contacts, which affect the amount of data that can be exchanged between two nodes. Even if the duration is known precisely, it may not be possible to predict the capacity due to fluctuations in the data rate. One approach to dealing with the capacity of contacts is that, if the volume of traffic is very small compared to the capacity of the contacts in the network, the capacity of the contact can be ignored, but in cases when the message is simply too large, it should be sent across the contact fragmentation. However, if the volume of traffic increases due to the increasing number of users, or due to large messages being exchanged, contact capacity becomes very important. In this situation, the best contact could become one that is “inefficient” according to other criteria but that has the largest contact volume and thus is best equipped to handle large traffic demands. Although the duration and capacity of contacts [18, 19] have been addressed, few of the routing strategies surveyed have considered delays caused by competing traffic. Exceptions are the EDLQ and EDAQ schemes proposed by Jain et al. [12], which compute the delay caused by waiting for competing traffic, then route the message on the path with the smallest delay.

Buffer Space

To cope with long disconnections, messages in DTNs must be buffered for long periods of time until an appropriate next hop is found, which means that intermediate relays must be equipped with buffers that have enough space to store all the messages that are waiting for future communication opportunities. Equipping nodes with buffers that can handle the entire demand is not feasible because of the nature of wireless mobile nodes. To deal with this issue, the available buffer space might need to be a consideration for routing strategies when they make decisions. In the studies surveyed here, all nodes are assumed to have an equal amount of buffer space, and decisions made by the routing strategies are not based on this resource.

Processing Power

One of challenges that affect DTNs is the amount of processing power in some devices, when one of the goals is to connect devices that are not served by traditional networks. These devices may have limited processing capability, in terms of CPU and memory. As a result, uncomplicated routing protocols must be run. The strategies presented in this proposal are not designed for extremely small devices such as sensors. Extensive survey of the difficulties associated with the power issues for routing in wireless sensor networks can be found in [20]. The routing strategies presented in this proposal are designed to deal with more powerful energy gateway nodes so that they are applicable for delay-tolerant sensor networks.

Energy

Limited energy supplies in some DTN nodes are a challenging problem because they are mobile and can not be kept connected to the power grid. Routing mechanisms consume a significant amount of energy by sending, receiving, and storing messages and by performing computations. Hence, routing protocols that perform fewer computations are more power efficient because they reduce power consumption. Routing strategies can also optimize power consumption by using energy-limited nodes sparingly. In DTN literature, many researchers have investigated general techniques for saving power in delay-tolerant networks [21]; however none of the routing strategies surveyed has incorporated power-aware optimizations. Thus, we will tackle this topic further in our future work.

2.1.2 Performance Measures

To compare routing strategies, three criteria have been chosen as a means of evaluating their performance: delivery ratio, latency, and number of transmissions.

Delivery Ratio

Delivery ratio is one of the most important metrics for evaluating DTNs because, in such networks, the network can be unable to deliver messages within an acceptable amount of time. In other words, messages are rarely lost but are subject to long delays. Thus, the delivery ratio is defined as the percentage of the total number of messages generated within a given time period that are delivered correctly to the final destination.

Latency

Latency represents the interval between the time a message is generated and the time it is received. This metric is especially important for applications that have a limited time window in which the data is useful. Although many applications can tolerate long waits, they can also benefit from short delivery latencies. An example is a scenario in which a DTN is used to deliver important messages, such as letters or emails to a mobile user; the messages must be delivered before the user moves out of the network.

Number of Transmissions

One of the protocol design goals in DTNs is to reduce the number of transmissions per message. The number of transmissions depends mainly on the decision strategy employed by the routing protocol when it chooses the next hop, either because of multiple copies of each message or because of protocol overhead. This situation results in some protocols transmitting more messages than others. An excessive number of message transmissions requires more computational resources, as some processing is required for each message, resulting in excessive energy consumption. The number of transmissions can therefore be used as a measure of the amount of contact capacity consumed by a protocol, as an approximate measure of the computational resources required, and as an approximate measure of power consumption.

2.2 Routing Strategies

Although a significant amount of work has been performed with respect to general DTN architecture [1], more effort and focus are still needed in order to reach an agreement about DTN routing algorithms, especially when for networks in which connectivity is neither enforced nor scheduled but is subject to predictions based on statistics about movement history of the nodes. DTN routing strategies can be classified into two categories according to the number of copies of the message and the information used to make decisions in order to find a destination: replication based, and forwarding, or knowledge, based. The following sections provide an overview of the most interesting algorithms, including their classification according to these two categories.

2.2.1 Replication (Flooding) Strategies

Many routing strategies in DTNs make multiple copies of each message in order to increase the chance that at least one copy will reach its destination or to reduce delivery latency. This strategy involves a clear trade-off between delivery time (latency) and resource consumption. Having more copies of the messages increases the probability that one of them will reach the destination and decreases the average delivery time. However, this process consumes a large amount of bandwidth, energy, and storage resources, which is proportional to the number of nodes in the network. The easiest approach is to send a single copy of the message in the network. However, if node failure is considered, this method will result in the message being lost. The most reliable approach is to have a controlled number of copies sent through the network, in order to balance the tradeoffs. A number of algorithms have been proposed for dealing with these issues. However, a protocol that will completely solve this problem has not yet proposed. This section gives an overview of the most interesting flooding-based routing protocols.

Tree-Based Flooding

In tree-based strategy, a message can be copied to a number of relays in order to increase the average delivery time. When a message is copied to a relay, an indication of how many copies the relay should make is included. This set of relays forms a tree of nodes rooted at the source. Many methods have been proposed for deciding how to make copies. A. Vahdat et al. [29] proposed a simple scheme that allows each node to make unlimited copies but that prevents the message from traveling more than a maximum of k hops from the source. This method limits the depth of the tree, but places no limit on its breadth. In [27] T. Small et al. proposed a scheme for limiting the node a maximum of L copies, which restricts both the depth and the breadth of the tree, the total number of copies being limited to a maximum of L. However, these schemes use a number of parameters that should be carefully tuned in order to obtain best performance. Groenevelt et al. in [5] proposed simple forwarding scheme called source forwarding (SF) or two hop forwarding. In this scheme, the source node forwards a message copy to the first $L-1$ nodes it encounters, and then each encountered node keeps a copy of the message until it meets the destination node of the message.

A more efficient alternative called Spray and Wait (S&W) is proposed by T. Spyropoulos et al. in [28]. It limits the total number of copies to L copies. In this scheme, a message source starts with L copies. When it encounters another node B with no copies, it distributes the responsibility for making $L/2$ of its current copies to B and keeps half for itself; when it has only one copy left, it switches to direct transmission. This scheme has been shown to be optimal if the inter-node contact probabilities are independent and identically distributed [52]. This scheme has been proven to perform a fewer number of transmissions with a competitive delay under network contentions such as limited buffer space and bandwidth. It also shows poor performance in real mobility scenario. This scheme is included in the evaluation of the proposed protocols.

Epidemic Routing

The fastest way to deliver messages is to spread the messages to all hosts, thus forming a type of persistent flooding, which is known as epidemic routing [29]. In this scheme, when two nodes encounter each other, they exchange all messages that they do not have in common. In this way, all the messages are eventually spread to all nodes in the entire network. Although this scheme is considered to be very robust against node failure and to provide the fastest message delivery, it is very wasteful of network resources. It has the highest number of transmissions, which increase rabidly as the number of nodes increases, which result in a very high energy consumption. It is also impractical to implement in most real wireless ad hoc networks in which bandwidth, buffer space, and energy are scarce resources. In such networks, epidemic routing causes a great deal of contention for limited buffer space and network capacity, resulting in large queuing delays and significant number of retransmissions and message drops [30, 43, 55]. These difficulties can degrade network performance dramatically.

A number of researchers have studied ways to improve the performance of the epidemic routing by reducing its overhead and quantity of resources consumption [31, 28, 101, 100, 23]. One challenging problem is finding a way to stop the propagation of a message through the network after it has been delivered. The authors in [32] proposed "death certificates" to solve this problem. A death certificate is a new message generated by the destination node and propagated through the network to inform nodes to delete the original message. Using a death certificate reduces the consumption of resources such as buffers and memory space. In addition, the death certificate is much smaller than the original message. Various schemes have been proposed for improving the propagation of the death certificates. In [27], the authors show that the more aggressive the death certificate propagation, the less storage is required at each node, while in [35], the authors show that more aggressive death certificate strategies lead to more transmitted messages. In [24], the authors examined a number of different schemes for suppressing redundant transmissions and cleaning up redundant messages from buffers after messages have arrived at their destinations.

To reduce the number of nodes that try to access a medium at the same time, the authors in [30, 33] applied a technique to forward a message to another node if it has a probability smaller than one, i.e., data is “gossiped” rather than flooded. While these protocols are more efficient than the original Epidemic Routing protocol, they still transmit many copies of each message.

2.2.2 Knowledge/ Forwarding Strategies

The strategies in this category require some knowledge of the network in order to select the best path, and the message is then handed over from relay to relay along this path. Forwarding strategies can be based on location information, the assigning of metrics to nodes or to links, or the use of opportunistic information based on the concept of history-based or utility-based functions. In other words, the forwarding decision is based on previous knowledge of the routes of potential carriers or on probabilistic approaches based on the history of encounters between nodes. The routing protocols presented in this proposal belong to this category.

Location-based Routing

The forwarding approach in this group of strategies attempts to make the least use of information about the network by assigning coordinates to each node. The coordinates can have physical meaning, such as GPS coordinates. A distance function is used to estimate the cost of delivering messages from one node to another. A number of approaches using GPS have been studied with respect to mobile ad hoc networks [36]. Alternatively, the coordinates can have meaning in network topology space, rather than physical space, a principle which has been used to estimate network latency between arbitrary nodes on the Internet [37, 96].

In general, location-based routing has two well-known problems. The first problem is that even if the distance between two nodes is small, there is no guarantee that they will be able to communicate due to obstruction that may exist [38]. The second problem rises from node movement. If a node moves, its physical coordinates change. If the network topology changes, a node's virtual coordinates change. In DTNs, due to the lack of an end-to-end path, the source is unable to update the coordinates of the destination node. These two problems make the implementation of location-based routing complicated. A number of studies have addressed these problems. Lebrun et al. [39] proposed using the motion vector of mobile nodes to predict their future locations. Their scheme passes messages to nodes that are moving closer to the destination, which results in a better delivery ratio a two-hop relay and requires less overhead than epidemic routing.

Leguay et al. [40] presented a virtual coordinate routing strategy called mobility pattern spaces (MobySpace), which is based on the use of a high-dimensional Euclidean space. The construction of MobySpace is based on the frequency of the nodes visiting each possible location. Each axis in Euclidean space represents a location, and the distance along the axis represents an estimate of the probability of finding a node at that location. In their strategy, the node coordinates are composed of a set of probabilities, and nodes that have similar probabilities of visiting a similar set of locations are more likely to encounter each other at a specific location. Further, the forwarding decisions rely on the notion that a node is a good candidate for taking custody of a message if it has a mobility pattern (based on constructed Euclidean space) similar to that of the message's destination. Routing is therefore performed by forwarding messages toward nodes that have mobility patterns that are increasingly similar to the mobility pattern of the destination.

Their results show that their approach outperforms epidemic routing. It consumes fewer resources than epidemic routing, while still delivering a substantial portion of the messages. However, neither of these studies addresses the problem of local minima or changing coordinates. They do show that these techniques are applicable in DTNs.

Gradient /Utility-Based Routing

An alternative approach is based on the use of utility functions that are calculated from an evaluation of context information, which is used as an assigned weight for each node in order for representing its suitability to deliver messages to a given destination. In particular, the history of the previous connections of the nodes [30, 68], the age of the last encounter timers [42], and probability of predicted contacts [43] are also used to calculate these functions. Other approaches used by Spyropoulos et al. in [28, 59]. He developed routing strategies using different utility routing metrics based on nodal mobility statistics , namely Most Mobile First (MMF), Most Social First (MSF) and Last Seen First (LSF). Ling et al. in [70] designed a feedback adaptive routing scheme based on the factors solely determined by the node mobility, where a node with higher mobility is given a higher factor, and messages are transmitted through nodes with higher influence factors. Some DTN message forwarding techniques, [71, 47, 69] have considered available bandwidth and buffer status in the routing metric to decide which message to replicate first among all messages in custodian buffer. The derivation of the routing metric, nonetheless, is not related to channel condition status.

Another scheme is called delegation forwarding [62], where a custodian node forwards a message copy to an encountered node if the encountered node has a better chance to "see" the destination. The key idea is that a custodian node (source or relay) forwards a message copy only if the utility function (represented by the rate of encounters between node pairs) of the encountered node is higher than all the nodes so far "seen" by a message, and then current custodian will update its utility value of that message to be equal to that of the encountered node. Mosli et al. in [65] introduced a DTN routing scheme using utility functions that are calculated from an evaluation of context information. The derived cost function is used as an assigned weight for each node that quantifies its suitability to deliver messages to an encountered node regarding to a given destination. Lindgren et al. in [43] introduced a routing technique in DTNs which takes advantage of the predicted encounter probability between nodes. Mosli et al. in [41] introduced a routing strategy based on the number of previous encounters to predict future contacts using Kalmin filter approach.

The techniques that based on number of previous contacts have two problems. One is multiple falsely detected contacts, as shown in Figure 2.1, when node D is within communication range of node C. Because node D may switch its power off and then switch it back on, node C will falsely detect more than one contact with node D. The same situation can occur when node$D$ exhibits an intermittent connection with node C, e.g., due to a communication barrier between them or the presence of node D on the edge of node C's communication range.

The other problem is related to permanent or quasi-permanent neighbors, as shown in Figure 2.1 when node A and node B move with the same velocity and in the same direction. Because no disconnection occurs for long time, only one contact between the two nodes would be counted irrespective of the long duration of the contact. On the other hand, both nodes encounter other nodes as they move, which can result in multiple contacts for each of them due to on and off links. A routing decision based on the number of contacts makes node$B$ a less suitable candidate for carrying a message for node A than other nodes that have a larger number of contacts, even though node B should actually be the preferred candidate for

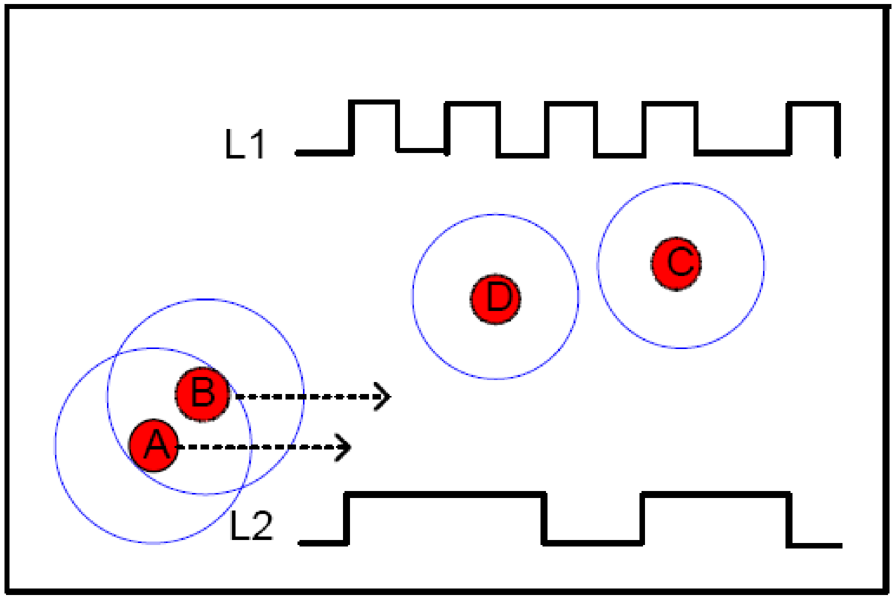


Figure 2.1: Example of the number of encounters in contact-based estimation

carrying the messages because it is in continuous contact with node A.

A sophisticated scheme was introduced by Spyropoulos et al., called Spay and Focus [28], which is characterized by addressing an upper bound on the number of message copies (denoted as L). In specific, a message source starts with L copy tokens. When it encounters another node B currently without any copy of the message, it shares the message delivery responsibility with B by transferring $L/2$ of its current tokens to B while keeping the other half for itself. When it has only one copy left, it switches to a utility forwarding mechanism based on the time elapsed since the last contact. This scheme has proven to significantly reduce the required number of transmissions, while achieving a competitive delay with respect to network contentions such as buffers space and bandwidth. This scheme is considered to be the best among the multi-copy utility forwarding approaches that have been proposed. The Spray and Focus approach is addressed in more detail below. An approach very similar to the Spray and Focus protocol was proposed by Li et al. [61], which differs from that by [28] in the employed utility function and queuing policy mechanisms. In specific, the utility function in is designed based on the probability of the duration of the contact time between pairs for a given time window interval. They compared their scheme to the Spray and Wait routing [28] and came to the same conclusion as T. Spyropoulos et al.[58]: if the community-based mobility model is used, the performance of the Spray and Wait protocol deteriorates.

In this proposal, each node must store a metric for all other nodes. Sufficient information must then be propagated through the network to allow each node to compute a metric for all destinations. The metric could be based on many parameters, such as the time of the last contact between the node and the destination, the history of the previous connections of the nodes, the remaining battery energy, or the mobility of a node. A variety of utility functions have been proposed for enhancing routing performance in DTNs because they have a significant impact on routing performance. The following is detailed discussion of two utility functions presentation, and the most effective utility-based routing protocols in DTNs.

Utility Function without Transitivity In a utility function scheme, nodes use information regarding the network obtained from last-encounter timers. These times are used as indirect information about the position of the nodes. This information becomes diffused throughout the network through the mobility process. In the majority of cases, a low encounter time for a given node implies that, the node is expected to be somewhere nearby. Therefore, a utility function can be defined based on these timers, within which a node can be examined with respect to its usefulness for delivering a message to another node. In [32, 43, 44], the authors used a gradient-based scheme to deliver a message to its destination. This scheme attempts to maximize the utility function for this destination.

Definition of Utility-Based Routing: Let every node X maintains a utility value $UX(Y)$ for every other node Y in a given network. The utility function X is a monotonic function of the respective last encounter timer $\tau x(y)$, where $\tau x(y)$ denotes the time elapsed since x last met y. Now let a node A carrying a message for a node D comes within transmission range of another node B, for which $UA(D) < UB(D)$. Then, forwarding the message to B results in a reduction of the expected delivery time of the message to its destination. The worst case would produce delay as large as that obtained by randomized routing.

Utility Function with Transitivity Even though the above scheme improves forwarding decisions, it suffers from a "slow start" phase, which is manifested more in large networks. In such a network, where the expected distance between a source and a destination is large, enough time is needed for the nodes near the destination to become "decoupled" and move near to the source [45]. These nodes will therefore not have a high enough utility to be chosen as next hops. Additionally, if it happens that some nodes around the message custodian last met the destination before the custodian did, the custodian will probably have to buffer the message and wait a long time until it moves into transmission range of the destination again, even if a path connected to the destination exists. This inefficiency is caused by the utility update rule of nodes, because each node updates its utility function for the destination only when it encounters that destination. As a result, location information takes a very long time to become diffused throughout the network, and by the time such information does become diffused, it is obsolete.

To deal with the drawbacks mentioned, Anders Lindgren et al. [43] proposed the use of "transitivity" when updating the utility function. When node A sees node$B$ often, and node B sees node C often, it can safely be inferred that A can be a good candidate to deliver a message to C (through B), even if A rarely sees C. Therefore, when A encounters node B, it should also update (increase) its utility for all nodes for which B has a high utility. Although the use of transitivity has improved the performance of forwarding decisions, the presentation of the transitivity function needs to be carefully chosen so that it actually improves performance, because the presentation should successfully capture the amount of uncertainty that is resolved with respect to the position of the destination node when it encounters a node that has more recent information for that destination. A number of the schemes proposed use different utility-based forwarding presentations. These schemes use a controlled number of message copies and are classified as hybrid–based approaches. The schemes related to the proposed work are addressed in more detail in the following section.

Spray and Focus Routing

The Spray and Focus scheme has been proposed as a method of overcoming the shortcoming of the Spray and Wait [28]. As in the spray and Wait scheme, in this scheme a fixed number of copies are spread initially in the spray phase, but then each copy focus phase is routed independently according to a utility function, i.e., looks for a node with a higher utility than its own. The utility function is based on the time elapsed since last encounters between pairs. The utility-based forwarding strategy provides very good performance in scenarios in which mobility is low and localized. This scheme has provided very competitive results in terms of delay in scenarios in which the Spray and Wait scheme loses its performance advantage, while making more transmissions per message compared to the Spray and Wait protocol [58].

Transitivity function

The Seek and Focus protocol employs the following transitivity function;

Definition: Let a node A encounters a node B at a distance dAB. Further let $\tau A(\bullet)$ and $\tau B(\bullet)$ denote the respective timer values that A and B have, respectively, for all other nodes. Finally, let $t_m(d)$ denote the expected time it takes a node to move a distance d for a given mobility model. Then

$$\vee_j \neq B : \tau_B(J) < \tau_A(J) - t_m(dAB),\ set\ \tau_A(J) < \tau_B(J) + t_m(dAB) \qquad (2.1)$$

Drawbacks of the Spray and Focus protocol

Although the Seek and Focus protocol has overcome some of the drawbacks in some of the routing protocols, it has its own limitations, as can be illustrated in the following scenario: Suppose a node A and a node B rarely encounter each other, e.g, they encounter each other on average every 90 time units. Then node A comes into the transmission range of node B at a distance dAB, and the time required for node B or A to cross distance dAB is 5 time units. Suppose that node A and node j usually encounter each other on average every 50 time units, and the time elapsed since they last encountered each other is 25 time units. Consider that node B encounters node j; i.e., they see each other on average every 20 time units, and it last encountered node j 10 time units previously. The utility transitivity function for node A would then be $\tau_A(j) = 5 + 10 = 15 < 25$, which means that the new value of $\tau A(j)$ is 15. That is, the utility function is improved although node A rarely encounters nodes J and B. As a result, in future, if any node encounters node A and it has a message destined for node j with a utility function value less than that of A, it will handover the message to node A even if node A is not a suitable candidate for delivering the message to node j. Moreover, the next time node A and node B may travel in different directions at different speeds. Consequently, the delivery time for a message would be increased. In addition, the calculation the optimum number of copies that can be used is still an issue, especially if a real mobility scenario is considered.

This research will propose other strategies that can use fewer copies than the Spray and Focus scheme by spreading a number of copies that is less than or at most equal to the number of copies used in the Spray and Focus scheme, while obtaining better guaranteed results than those of other schemes described in the literature. Chapter 4 introduces the new utility update rule along with the proposed forwarding strategy, or routing algorithm. The proposed update rule captures the importance of history of encounters and how utilizing it can improve the forwarding strategy.

Social Networks based Forwarding

Some studies have investigated the impact of human mobility and their (potential) social relations on the design and performance of the appropriate routing algorithm [91, 90, 89, 88, 94, 49]. The forwarding approach in this group of strategies is called social network based forwarding. With these schemes, the variation in node popularity, and the detectability of communities, are employed as main factors in forwarding decisions. Hui et al. in [94, 90] investigated the human mobility traces in terms of social structures, and use these structures in the design of forwarding algorithms for Pocket Switched Networks (PSNs). They proposed a social based forwarding algorithm, called BUBBLE, which is shown empirically to improve the forwarding efficiency significantly compared to oblivious forwarding schemes. Daly et al. in [88] presented social network analysis metrics that may be used to support a novel and practical forwarding solution to provide efficient message delivery in disconnected delay-tolerant MANETs. These metrics are based on social analysis of a node's past interactions and consists of three locally evaluated components: a node's "betweenness" centrality (calculated using ego networks), a node's social "similarity" to the destination node, and a node's tie strength relationship with the destination node.

In [93, 65] the authors extended the operational properties of utility functions to also predict future attributes of potential message carriers; the new notion, which includes both the utility functions and its predictability extensions is the context. The evaluation of a node's context is made based on two main criteria, namely the rate of change of connectivity of the host (i.e., how possible it is that this node will move and meet other nodes) and the energy level (i.e., how long will the node stay "on", so that it will be able to interact with encounters). The analysis is based on the fact that "mobile networks are social networks after all, since mobile devices are carried by individuals" . The authors report acceptable delivery ratios with relatively low delivery delays and small overheads.

2.3 Collaborative Reinforcement Learning (CRL)

CRL is used for tackling the complex time-varying problems where global knowledge on system behaviors is not available [75, 79]. With a CRL coordination model, the agents can cooperate with each other to solve a system-wide optimization problem that could be composed of a set of discrete optimization problems (DOP). An agent can solve one or a number of discrete optimization problems via reinforcement learning by exchanging some key information with neighbor agents, which further contributes towards the solution of the system-wide optimization problem. An individual agent only possesses partial knowledge about the system-wide state and knowledge about their neighbors. As a result, each agent serves as a member of the dynamic population that joins (or leaves) the system by autonomously establishing (or tearing down) connections with their neighbors without making any use of system-wide knowledge.

With CRL, the path selection is based on the expected performance of an agent starting with initial state, s, in which the algorithm exercises an optimal state transition policy thereafter. An estimated value function $V(s)$ is employed at a CRL agent as the cost function in solving a DOP. $V(s)$ can also be presented as an optimal action-value function, $Q(s,a)$, and their relation can be expressed as

$$V(s) = max_a Q(s,a)$$

Two transition states are identified in CRL; local on the current agent n_i, and remote to a neighboring agent n_j. The estimation of the cost of transition from the local to the remote state takes into consideration the connection cost between the current agent and the neighboring agent. Therefore, the estimated optimal action value function, $Q_i(s,a)$, should includes both the value function for the state $V_j(s')$ that is received from the neighboring agent, and the connection cost of the state, $D_i(s',a,s)$. The connection cost for a transition from the local state of current agent to the remote state at a neighboring agent should reflect the underlying network cost as well as the cost of transferring control from the source agent to the target agent. The transfer of control involves terminating the DOP at the originating agent and start solving a new DOP at the target agent. The cost function is given by:

$$V(s) = R(s,a) + max_a \sum P_i(s' \mid s,a)$$

$$. \left(D_i(s' \mid s,a) + Decay(V_j(s'))\right),$$

$$V(s) = max_a[Q_i(s,a)] \tag{2.2}$$

where, $P_i(s'|\ s,a_j)$ represents set of transition models that describe the probability of making state transition from state s to state s' under delegation action a, $D_i(s' \mid s,a)$ is the estimated connection cost model at agent n_i of making a transition from sate s to state s' under delegation action a, $Decay(Vj(s'))$ is the decay model used at agent n_i to decay the V values of last advertised costs to given destination agents. This mechanism is used to eliminate and degrade agents that have a lower contact frequency with node n_i. Here, $R(s, a)$ is the Markov Decision Process(MDP) termination cost.

To overcome the lack of prior or centrally managed knowledge on the network environments, a CRL has been proposed [75] for MANETs, which is characterized by an autonomous and self-organizing design for developing MANET routing protocols. The technique has been proven as a successful implementation regarding robustness and scalability in the context of MANET routing. Inspired by this, we envision that the concept of CRL can also be applied to the DTN routing protocol design in spite of the decentralized and intermittently connected characteristic in DTNs; and the use of a CRL model should be able to improve the performance and scalability of the DTN routing protocol operations.

2.4 Buffer Management

While there is considerable amount of effort for improving routing techniques in DTNs [98, 99, 52], many of them have not considered the fact that each node could be a hand-held and battery-powered device with stringent power consumption constraint and buffer size limitation. The buffer limitation may cause message drop/loss due to buffer overflow, and the insufficient link bandwidth during a contact and short duration of a contact may not allow the node to transfer all the intended messages. This causes a big challenge in the implementation of some previously reported schemes. To address this issue, a few studies have examined the impact of buffer management and scheduling policies on the performance of DTN routing [95].

Lindgren et al. in [77] evaluated a set of heuristic buffer management policies based on locally available nodal parameters and applied them to a number of DTN routing protocols. The following queue management policies are used to decide which message should be dropped if the buffer is full when a new message has to be accommodated.

- FIFO – First in first out. Handle the queue in a FIFO order. The message that was first entered into the queue is the first message to be dropped.
- Drop Random(DR): the selection of message to be dropped is random.
- Drop –Least-Recently-Received (DLR): the message with the long stay time in buffer will be dropped. The idea is that the packet with in buffer for long time has less probability to be passed to other nodes.
- Drop-Oldest (DOA): the message with the shorted remaining life time (TTL) in network is dropped. The idea of dropping such packet is that if packet TTL is small, it is in the network for long time and thus has high probability to be already delivered.
- SHLI – Evict shortest life time first. In the DTN architecture [1], each message has a timeout value which species when it is no longer useful and should be deleted. If this policy is used, the message with the shortest remaining life time is the fist to be dropped.
- DL-Drop last(DL): it drops the newly received message.
- Drop front(DF): the message that enters first in the queue is dropped first.
- MOFO – Evict most forwarded first: In an attempt to maximize the dispersion of messages through the network, this policy requires that the routing agent keeps track of the number of times each message has been forwarded. The

message that has been forwarded the largest number of times is the first to be dropped, thus giving messages that have not been forwarded fewer times more chances of getting forwarded.

Zhang et al. in [33] addressed this issue in the case of epidemic routing by evaluating simple drop policies such as drop-front and drop last, and analyzed the situation where the buffer at a node has a capacity limit. Stylianos Dimitriou et al. [83] proposed buffer management policy based in using two types of queues for two types of data traffic; a low-delay traffic (LDT) queue and a high-delay traffic (HDT) queue. Wahidabanu et al. in [84] proposed an approach based in classifying the bundles into classes of services, and the main buffer is divided into queues accordingly. Separate queue is maintained for each class of service, and the bundles are scheduled according to the class of service. Noticeably all the above mentioned policies based only on the local knowledge of some network information.

Khrifa et al. in [76] proposed an interesting approach for solving the problem of buffer management by way of a drop policy and a scheduling scheme. This is the first study that explicitly takes global knowledge of node mobility as a constraint in the task of buffer management. Specifically, their method estimates the number of copies of message i (message under consideration) based on the number of buffered messages that were created before the message i. Although interesting, the method may become inaccurate when the number of network nodes is getting larger, especially for newly generated messages. Meanwhile, the effect due to the change of the number of message copies during the remaining lifetime of a message is not considered in the utility function calculation. This means the utility function is only affected by the current message copies and its remaining lifetime. It is clear that the above mentioned studies leave a large room to improve, where a solution for DTN buffer management that can well estimate and manipulate the global status is absent.

2.5 Summary and Observations

In this chapter we have surveyed existing techniques for routing and buffer management in DTNs. While a wide variety of methods address the routing problem, they can be classified according to two key properties: replication based and knowledge based. Protocols are subcategorised according to these classifications, and their advantages and limitations have been highlighted as well.

The survey and classification led to the following observations. First, achieving a low delivery delay and a high delivery ratio with low resource consumption requires techniques that rely on knowledge about both the topology and replication. This concept has been implicitly noted by several of the researchers in the field. Thus, the challenge is to determine the correct balance between redundancy and resource consumption, and to find manageable solutions for using network topology information. These issues are still open and need to be solved. Second, in cases in which message volume is low, simple epidemic routing works extremely well, which suggests that small experimental deployments could be rapidly developed based on epidemic routing, allowing researchers to work with actual network topology and traffic data, which could then be used to design new routing strategies. Third, although some of routing schemes have been reported to deal with the limited network resources, none of them have investigated thoroughly how the protocol should take advantage of dynamic network status to improve the performance, such as packet collisions, wireless link conditions, and nodal buffer occupancy. There is obviously some room to improve for the multi-copy routing schemes in the DTN scenario considered in this study.

With this in mind, the main feature of the proposed protocol should be the strong capability in adaptation to the fluctuation of network status, traffic patterns/ characteristics, and user behaviors, so as to reduce the number of transmissions, message delivery time, and increase delivery ratio. This can be achieved by jointly considering node mobility statistics, congestion, and buffer occupancy, which are subsequently fused in a novel quality-metric function. Fourth, although some effort has been reported to improve data delivery by proposing some buffer management policies, most of existing policies are based on the local network state. Efficient policies should take in the consideration the global network state in the consideration in order to make efficient drop or forwarding decision such that the delivery delay is reduced and/or the delivery ratio is improved.

Chapter 3

Problem Description and Solution Approach

The efficiency of routing protocols in DTNs is affected by many factors, such as energy consumption, buffer space, fault tolerance, and bandwidth. Most of DTN protocols work under the assumption that all nodes in the network have no contention (infinite buffer space and unlimited bandwidth). On one hand, flooding-based (or multiple-copy based) schemes are robust and provide short delays. However, they consume large amounts of energy, bandwidth, and memory space. In addition, they suffer from contention under a high traffic load, during which packet drops can significantly degrade performance and scalability [43, 30, 25, 6, 97]. Consequently, these drawbacks may render such algorithms prohibitive for bandwidth-constrained and energy-constrained applications, which is a common case in a MANET. Therefore, the main goal of this research is to design a framework of utility-based (hop-by-hop) routing mechanisms that achieves both minimum delay and low consumption of bandwidth, energy, and memory in large-scale intermittently connected mobile networks. Achieving this goal, begins with a thorough designing of multi-copy routing mechanism for intermittently connected mobile networks under no contention.

This design is taken as a building block.

From this building block, the work is extended to study the same problem while considering designing multiple-copy routing approaches that can deal with networks under contention. Improvement in the performance of the utility-based routing protocols in DTNs is mainly due to the forwarding policy applied when two nodes are within transmission range of each other. The forwarding policy is further guided by the utility function, which adapts to the network state such as available bandwidth, buffer occupancy status, and the historical data of nodal interconnectivity. It is expected that the presentation and information updates used for calculating the utility function will have significant impact on the performance of the protocol. In general, this research addresses three key factors in the design of a large-scale DTN: connectivity, contact capacity, and limited buffer capacity. These factors are described in the following subsections.

3.1 Major Issues in Encounter-Based DTNs

3.1.1 Network Connectivity

The nature of the underlying network connectivity plays an important role in protocol design. A message, which is a trunk of data that can be up to several megabytes, might be buffered at a custodian node until a next hop is found. The buffer time of a message could vary from seconds to days. Thus, an important consideration is the nature of the underlying network connectivity, which depends strongly on application scenarios and environments under consideration. It has been proven that user mobility serves as fundamental input into the performance of a routing protocol in DTNs, and the mobility model adopted for the performance analysis will have a significant impact on the results [54]. Simply adopting the RWP or the RW model can never effectively resemble the actual behavior of human activities due to the unrealistic assumption that all nodes are regulated under identical and independent distributed movements [53, 54].

The aim of this research is to study possible actions when connectivity is subject to the statistics of a mobility model followed by nodes. Providing an efficient solution for the routing problem requires consideration of a mobility model that resembles the real behavior of human movements. Nodes in networks with this type of mobility follow some types of human activities that have predicted schedules. These activities can be represented as contacts that can predictably be brought up or down. It can be argued that a promising strategy is to employ transitivity function that can aggregate the information about network connectivity and use it to make an intelligent forwarding decision. This process ensures that each node in the network can efficiently build useful knowledge about all the other nodes.

3.1.2 Contact Capacity

In DTNs, the most stringently limited resources are the available bandwidth and the duration of the contacts. When the transmission range is fixed, there is no way to increase the link capacity [56, 57], which affects the amount of data that can be exchanged between two nodes. Even if the duration is known precisely, it may not be possible to predict capacity due to fluctuations in the data rate. The contact capacity becomes very important if the volume of traffic increases due to an increasing number of users, or due to large messages being exchanged. To deal with this issue, the designed protocol should reduce the number of transmissions per message in order to avoid violating the bandwidth constraint. This step requires the implementation of an efficient forwarding strategy that guarantees end-to-end forwarding with the least number of transmissions and also guarantees that the least number of message copies will be spread in the case of multiple-copy routing with low overhead.

3.1.3 Buffer Space

To achieve data delivery in a DTN, a node may carry a message for a long period of time, until it encounters another node with a higher forwarding opportunity. Additionally, multiple copies of a message are often propagated in order to increase the probability of delivery. This combination of long-term storage and replica imposes a high storage overhead on relay nodes. Thus, efficient buffer management policies are necessary for effective decisions about messages should be kept or discarded when the buffers of nodes are operating close to capacity limit.

3.2 Routing Problem in Encounter-based DTNs

There are many routing protocols have been designed for encounter-based DTNs environment [60] – [62]. The main effort in developing these routing schemes is finding a method of exploiting the nodal mobility in order to predict the future contacts between a pair of nodes that is further employed as a main factor in the message forwarding decision. A number of the techniques that have been proposed for DTN routing use different criteria for predicting future contacts, e.g., the idea that a most recently seen node is more likely to be met [9, 102]. Some schemes [46], the ones that are compared with the proposed routing protocols, use a utility function presentation implicitly based on the distance between encountered nodes. However, this presentation has its own drawback, as appointed in Chapter 2.2.

Some techniques rely on the assumption of a predefined rarely changing movement pattern in which routing decisions are based [5]. However, it is not clear how to determine mobility patterns. Previous techniques [41, 43] predict future contacts based on the number of previous contacts. Predicting contacts is critical since it determines the suitability of the encountered node as the next carries of the message. Such an approach has two critical problems, as appointed in chapter 2.2.

More importantly, most of previously reported utility-based forwarding schemes assume that each node has sufficient resources for message buffering and forwarding. None of them have investigated how the protocol should take advantage of dynamic network status to improve the performance, such as packet collisions, wireless link conditions, and nodal buffer occupancy. The performance of the existing protocols degrade dramatically when the traffic demand is high in a network with limited resources.

With these drawbacks in mind, We set two routing objectives. First, design and employ an accurate and efficient nodal mobility exploitation method represented in a from of a utility function. Second, design DTN routing protocols that can deal with different network capacities. Our design is based on the assumption of the existence of two types of network environments; i) a network with sufficient resources to handle traffic demand, and ii) a network with insufficient resources such as buffer space, and bandwidth, which have major affect on messages forwarding in DTNs.

For the first environment (a network with sufficient resources, or lightly loaded network), an encounter-based protocol, called Self Adaptive Routing Protocol (SARP) is introduced [63]. The introduced solution has the goal of investigating the effect of deploying a self-organized framework for routing messages in sparsely connected mobile networks. SARP can achieve minimum delivery delay, high delivery ratio, and low transmissions. SARP alleviates the shortcomings of existing utility-based protocols in networks with considerably sufficient resources.

For the second environment (network under contention), two contention aware routing techniques are introduced, Self Adaptive Utility-Based Routing Protocol (SAURP) [85], and Adaptive Reinforcement-Based Routing Protocol (ARBRP) [64]. Each protocol employing a different way of exploiting the network state information and the nodal mobility. SAURP uses a utility function in a form of inter-contact time as the main factor on its forwarding decision, while ARBRP employing a utility function in a form of contact time duration as the main factor on its forwarding decisions. The main feature of the introduced protocols is the strong capability in adaptation to the fluctuation of network status, traffic patterns/characteristics, and user behaviors, so as to reduce the number of transmissions, message delivery time, and increase delivery ratio. This is achieved by jointly considering node mobility statistics, congestion, and buffer occupancy, which are subsequently fused in a novel quality-metric (utility) function. In specific, the link availability and buffer occupancy statistics are obtained by sampling the channels and buffer space during each contact with another node. The developed quality-metric function targets to facilitating decision making for each active data message, resulting in optimized network performance.

In summary, using an efficient presentation of utility function values along with a smart forwarding strategy can reduce the number of transmissions, delivery delays, and increase the delivery ratio in the network, which is the main motivation behind this research. Other challenges, such as dealing with node failure and meeting delivery deadlines for some applications, are still unresolved. One possible solution for dealing with these issues is to spread multiple copies of each message in the network. However, spreading many copies of messages throughout the network creates other problems, such as buffer space and bandwidth violations. Developing a multi-copy routing protocol that uses a low number of transmissions combined with an efficient use of buffer space is another motivation of this work.

3.3 Buffer Management and Scheduling Problem in DTNs

Most DTN routing protocols have assumed unlimited contact bandwidth and negligible storage overhead [29, 52, 102] without considering that each node could be with a limited buffer space and contact bandwidth. Note that buffering and forwarding unlimited number of messages may also cause intolerable resources and nodal energy consumption; and it is imperative to set up bandwidth and buffer limitations at the DTN nodes to better account for the fact that each node could be a hand-held and battery-powered device with stringent contact capacity and buffer size limitations. With such buffer and bandwidth limitations at the DTN nodes, message drop/loss could happen due to buffer and/or traffic overflow. This leads to a big challenge in the implementation of most previously reported schemes such as those belonging to the class of epidemic (flooding) routing. In order to facilitate the implementation of such routing schemes, the following issues should be solved:

- (Scheduling issue due to limited contact capacity) What is the decision that should be made about which message should be forwarded first to node A among all messages in node $A's$ buffer, so as to maximize the global delivery ratio and/ or minimize the delivery delay of all messages in the network?.

- (Buffer management issue due to limited buffer capacity) What is the decision that should be made about which message should be dropped among all messages stored at node $B's$ buffer if the decision is made (taken by routing algorithm) to forward message i from node A to node B and $B's$ buffer is full in order to maximize the global delivery ratio and/ or minimize the delivery delay of all messages in the network?.

Part of this thesis addresses the problem of buffer limitation by introducing novel buffer management and scheduling framework for DTNs considering two different aspect of routing families [82]: flooding-based routing, where the Epidemic [29] and controlled flooding (source forwarding) [5] schemes mechanisms are considered in the formulation, and encounter-based routing, where SAURP [85] mechanism is considered in the formulation. The proposed buffer scheduling policies are aiming to enable an effective decision process on which messages should be dropped or forwarded when the buffer is full. Such a decision is made by evaluating the impact of dropping each buffered message according to collected network information.

3.4 Summary

This chapter has presented the problem statement. It identifies the main problems with the existing encounter-based routing protocols and buffer management strategies and suggests a framework for developing routing protocols and buffer management strategies that can be integrated into DTNs. Next chapter introduces the first step of our effort of solving the routing problems in DTNs.

Chapter 4

Proposed Framework for Self-Adaptive Routing

This chapter explores the problem space of multi-copy routing in encounter-based DTNs and presents the proposed adaptive encounter-based routing protocol for tackling the multi-copy routing problem.

4.1 System Model

For the purpose of this research, a network consists of a number of nodes moving independently on a $\sqrt{N} X \sqrt{N}$ 2-dimensional torus in a geographical region, as shown in Figure 4.1. The simulation area is divided into a number of communities. The movements of nodes are according to the community-based mobility model [50].

Each node can transmit up to distance $K \geq 0$ meters away, where $K/\sqrt{N}$ is smaller than the value required for connectivity [60], and each message transmission takes one time unit. Euclidean distance is used to measure the proximity between two nodes (or their positions) A and B. Additionally, we assume that the network is disconnected at most times, and that the transmission is faster than the node

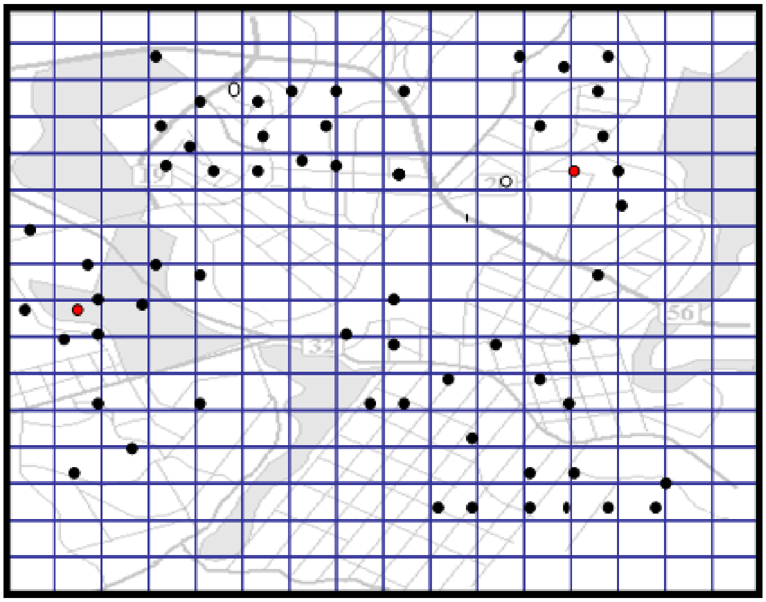


Figure 4.1: The mobility space

movement, which is a reasonable assumption with modern wireless devices [45].

4.2 Mobility Model

Popular mobility models such as RWP and RW that have been employed in DTNs research [48, 50] do not hold if the actual behavior of real-life situations is considered, e.g., university campuses, conferences, work, etc. These models assume that nodes are uniformly distributed in the network area and that all nodes have equally frequent movements to every network location. The mobility characteristics of the nodes are also considered to be the same: every node's mobility process is identical and independently distributed from all others. Because of the limitations of these two assumptions, a number of research studies have been conducted based on real-life networks. These studies have motivated us to use mobility model that can better resemble real node movements called the Community-Based Mobility Model [50].

4.2.1 Community-based Mobility Model (CMM)

The model is based mainly on two states: local and roaming [50]. Nodes in this model belong to different communities in the network and fluctuate between these two states. The movement inside the network is described as follows: (i) The node's movement inside a community consists of local and roaming epochs. (ii) In a local epoch, a node performs movements in random directions only within the node's local community. (iii) In a roaming epoch, a node performs random direction movement inside the entire network (or performs restricted movement to predefined communities). (iv) After performing the movement, each node calculates its next epoch as follows: If the node's previous epoch was local, its next epoch is local with a probability of p_l, or roaming with a probability of $1-p_l$. (v). If the node's previous epoch was roaming, the next epoch is roaming with a probability of p_r, or local with a probability of $1-p_r$. Tuning the parameters allows the generation of a large number of scenarios using the community-based mobility model. In this work, we generate a specific scenario closely resembling the reality of an actual network. In this scenario, different nodes share specific communities, such as several library buildings or offices on a campus, several offices in a company, or several entertainment venues at one location. This scenario is more realistic compared to other scenarios in which all nodes move uniformly at random throughout the entire network. With this model, the proposed protocols can be examined with respect to different networks sizes, different ranges of connectivity, and different traffic loads.

4.2.2 MAC protocol

In this study, a shared channel based on a simplified version of the slotted collision avoidance MAC protocol with Clear-to-Send (CTS) and Request-to-Send (RTS) has been implemented in order to arbitrate between nodes contending for the shared channel. In the proposed implementation, we assume that each message takes one time unit to be transmitted (unless it is specified). When a node receives a message, it sends a small acknowledgment packet (ACK) back to the sender.

4.3 The Self-Adaptive Routing Protocol (SARP)

The main aim of the proposed Self-Adaptive Routing Algorithm (SARP) is to adapt itself according to network behavior in order to reduce the number of transmissions, the delivery time, and increase delivery ratio. These improvements can be achieved by employing a more efficient updating strategy for the stochastic information at each node. Although the idea of using transitivity itself is not new [43], a transitivity function and an inter-contact time table are presented in a different way so that the proposed utility function presentation is more efficient. It comprises both contact time duration and encounter rate, that is, the number of encounters that nodes have during a time window. More insight about the metrics used by the proposed protocol is provided in the next subsection.

4.3.1 Prediction of Future Contacts

To address the problem, the proposed routing is based on an inter-contact time which intrinsically relies on the duration and frequency of previous contacts, rather than only the number of previous contacts. Including the total duration of all the contacts as the parameter is expected to better reflect the likelihood that nodes will meet with each other. Without loss of generality, consider two nodes, A and B. At any time, each node broadcasts a pilot signal each k time units in order to look for its neighbors, the nodes within its transmission range. To consider a contact as an encounter, the duration of the contact and the time duration between two consecutive pilot signals should be at least equal to the time needed to transmit one message. When A encounters B, for time duration T_{AB}, each time the pilot signal of A finds that B is still within A's transmission range, it increases the number of encounters between A and B by 1. The number of encounters during one contact is calculated by

$$n_c = \left\lfloor \frac{T_{AB}}{T_p} \right\rfloor \tag{4.1}$$

where T_p is the time duration between two consecutive pilot signals. Regardless of the time synchronization and the duration of the time that nodes A and B stay connected to each other, $T_{AB} = T_{BA}$. The average of the inter-contact time of a future contact between nodes A and B is estimated approximately as follows. Consider that A and B encounter each other, and that so far they have encountered each other n times (including the new number of encounters resulted form the new contact). The inter-contact time between A and B is then calculated as:

$$\triangle T_{(A,B)} = \frac{t}{n} \tag{4.2}$$

where t is the time at which nodes A and B move out of transmission range of each other. The average inter-contact frequency describes how often the two nodes encounter each other per unit of time. This type of information is considered useful for constructing a nodal mobility model based on the historical behavior of each node, where the encounter frequency of each pair of nodes is taken as an abstract of the real mobility model. In this technique, the inter-contact time is computed over the entire history of a node, which is easy since the computation requires only the division of current time over the sum of the number of contacts, from the initial time when the observations began.

Good performance can be further achieved by setting sliding window parameters so that each node can accumulate enough information about all other nodes. Thus, in the proposed model, a node will maintain a list of encountered nodes in a specific time window. The encounter frequency information for each pair of nodes is calculated, maintained, and exchanged among nodes, which serves as important input to the message forwarding decision-making process for each contact. The message-forwarding strategy, including weighted copy, transition, and updating rules, is presented in the next subsection.

4.3.2 Forwarding Strategy in the SARP

The forwarding strategy is based on the concepts described below.

- Each node keeps track of the history of the average time rate of the inter-contacts with all the other nodes in the network, that is, how often nodes come into transmission range of each other. For example, a node A encounters node B every R time units. The list is called inter-contact table.

- Each node maintains a list of nodes that frequently encounters in a specific time window. The list is maintained through a table that contains the history of the average inter-contact time of each node in the list for every other node. The list is called history table.

- At the beginning, the history of the contents of inter-contact table is set to infinity.

- When two nodes encounter each other, the timer is set to the current time, and the timer starts counting the elapsed time until they move out of transmission range of each other. The number of encounters during this new contact as given by equation (4.1) is added to the number of current encounters. The average of the inter-contact time rate is calculated as given by equation (4.2).

- A node in the history table of a custodian node or in that of an encountered node can take part in a routing decision only if its inter-contact time value regarding the destination is better than those of custodian node or the encountered node respectively.

- **The Weighted Copy Rule**

The source of a message initially starts with L copies; any node A that has $N_A > 1$ message copy tokens (source or relay) and that encounters another node B with no copies hands over to node B a number of copies according to its goodness for the destination node D. Node A hands over N_B of the message copy tokens to node B and keeps the rest for itself according to the following formula:

$$N_B = \left\lfloor N_A \left(\frac{\triangle T_{(A,D)}}{\triangle T_{(B,D)} + \triangle T_{(A,D)}} \right) \right\rfloor \tag{4.3}$$

where N_A is the number of message tokens that node A has, $\triangle T_{(B,D)}$ is the inter-contact time rate between node B and node D, and $\triangle T_{(A,D)}$ is the inter-contact time between nodes A and D. This formula guarantees that the largest number of message copies is spread to relay nodes that have better information about destination node. After L messages have been copied to custodian nodes, each of the L nodes carrying a copy of the message performs according to the forwarding rule as descried later in the following subsection (4.3.3).

The Updating Rules

Two types of updating rules are identified: The decay rule and the transitivity update rule:

(1) The Decay Rule

When nodes do not encounter other nodes for a while, they are less likely to be good candidates for forwarding messages to each other; thus, the frequency of the inter-contact time rate should be reduced (aged). The aging equation is as follows:

$$\triangle T_{(A,B)} = \frac{t}{n} \tag{4.4}$$

where t is the current time, n is the number of times nodes A and B encountered each other since the start time at t = 0. This rule is applied before any message forwarding.

(2) The Transitivity Updating Rule

The updating rule also has a transitivity property based on the observation that if Node A frequently sees node B, and node B frequently sees node D, then node A has a good ability to forward messages destined for node D through B. We formulated the updating rule as follows:

$$\triangle T_{(A,D)new} = \alpha \triangle T_{(A,D)} + (1-\alpha)(\triangle T_{(A,B)} + \triangle T_{(B,D)}) \tag{4.5}$$

where α is weighting factor that must be less than 1 to be valid.

$$\alpha = \frac{\triangle T_{(A,B)} + \triangle T_{(B,D)}}{\triangle T_{(A,D)}}, \ \triangle T_{(A,D)} > \triangle T_{(A,B)} + \triangle T_{(B,D)} \tag{4.6}$$

α has a significant impact on the routing decision rule. From theoretical perspective, when a node is encountered that has more information for a destination, this transitivity effect should successfully capture the amount of uncertainty to be resolved regarding the position of the destination. Thus, a transitivity property is needed to update values only when $\triangle T_{(A,D)} > \triangle T_{(B,D)}$ in order to ensure that node A reaches D through B. Otherwise, if $\triangle T_{(A,D)} < \triangle T_{(B,D)}$, the transitivity property is not useful since node A is a better candidate for forwarding messages directly to node D rather than forwarding them through B. This rule is applied after nodes finish exchange messages.

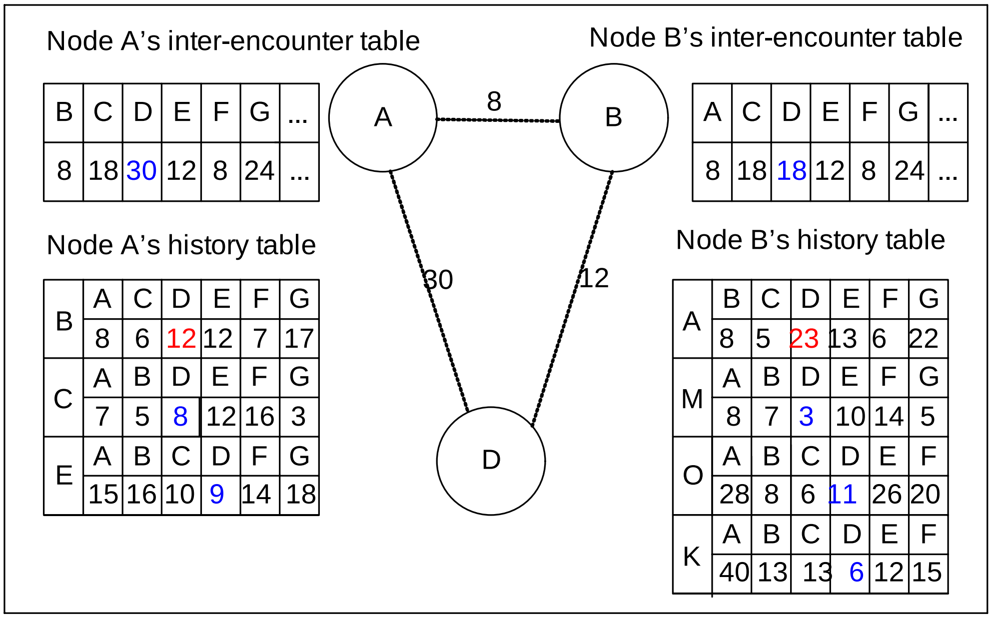


Figure 4.2: Illustration of the transition rule

The Transition Rule

The following example illustrates the concept: Suppose two nodes A and B encounter each other at time t, and node A has a message destined for node D. As mentioned, each node maintains a history of encountered nodes; i.e., for nodes A and B, in addition to their inter-contact time tables, they maintain tables of the history of the inter-contact time tables for other nodes that frequently being encountered in a specific time window. The table is as shown in Figure 4.2.

The terms required for calculation are defined as follow:

- $R1$ = the average of the inter-contact time of node D in node A's history list and inter-contact table.

$$R1 = \frac{\left(\triangle T_{(A,D)} + \triangle T_{(C,D)} + \triangle T_{(E,D)}\right)}{3} \tag{4.7}$$

- $R2$ = the average of the inter-contact time of node D in node B's history list and inter-contact table.

$$R2 = \frac{\left(\triangle T_{(B,D)} + \triangle T_{(M,D)} + \triangle T_{(K,D)} + \triangle T_{(O,D)}\right)}{4} \tag{4.8}$$

- $\triangle T_{(A,B)}$ = the inter-contact time between nodes A and B.
- $\triangle T_{(A,D)}$ = the inter-contact time between nodes A and D.
- $\triangle T_{(B,D)}$ = the inter-contact between nodes B and D.

Four cases can be identified:

Case1: If $R1 > R2$, $\triangle T_{(A,D)} > \triangle T_{(B,D)}$ node A hands over the message to node B.

Case2: If $R1 > R2$, $\triangle T_{(A,D)} < \triangle T_{(B,D)}$ the roulette wheel selection is applied on the probability after the tables for A and B are updated. The probability of handing over the message to node B,

$$P_b = R1/(R1 + R2) \tag{4.9}$$

The probability of keeping message with $A = 1 - P_b$.

Case3: If $R1 < R2$, $\triangle T_{(A,D)} > \triangle T_{(B,D)}$ the same as in case 2.

Case 4: If $R1 < R2$, $\triangle T_{(A,D)} < \triangle T_{(B,D)}$ node A keeps the message, since so far it is the best candidate to deliver the message to node D. Applying this mechanism is very useful, especially when the amount of uncertainty regarding to the destination for node A is very high.

4.3.3 The Forwarding Rule

- At the beginning, all nodes' tables are not useful, i.e., above a predefined threshold value $Th1$. Thus, a relay node can use the randomized routing to for message handover.

- If the destination node is one hop away from an encountered node, the custodian node hands over the message to the encountered node.

- If the inter-contact rate value of the encountered node relative to that of the destination node is less than $Th1$ and less than that of the custodian node by a threshold value $Th2$, a custodian node hands over the message to the encountered node. Otherwise, it applies its transition rule to hand over the message. The flowchart of the mechanism of the algorithm is shown in Figure 4.3.

4.4 Performance Evaluation

To evaluate the performance of the proposed protocol, extensive simulations were performed for variety of scenarios. In order to conduct the evaluation, a DTN simulator, a discrete-event simulator for delay tolerant networks was created. It is based on the simulator used in [52, 51]. SARP was compared with previously reported approaches under different variations of buffer capacities, traffic loads, and connectivity levels, using a realistic mobility scenario. The comparisons are in terms of the average delivery delay, the total transmissions, and delivery ratio.

4.4.1 Community-Based Mobility Scenario

This study uses Community-Based Mobility Model [52, 50] which is known to well resemble real node movements. In this model, the simulation area is divided into small communities, and each node has its own community. Each node may have a preferred community that it visits frequently. It may move preferentially for the majority of time, leaving its community and roaming into other communities for some time, then returning to its home community. Each node may also have different mobility characteristics in addition to different communities. Some nodes may spend most of their time inside their community, while others may be more mobile and roam from one community to another. The community-based model allows

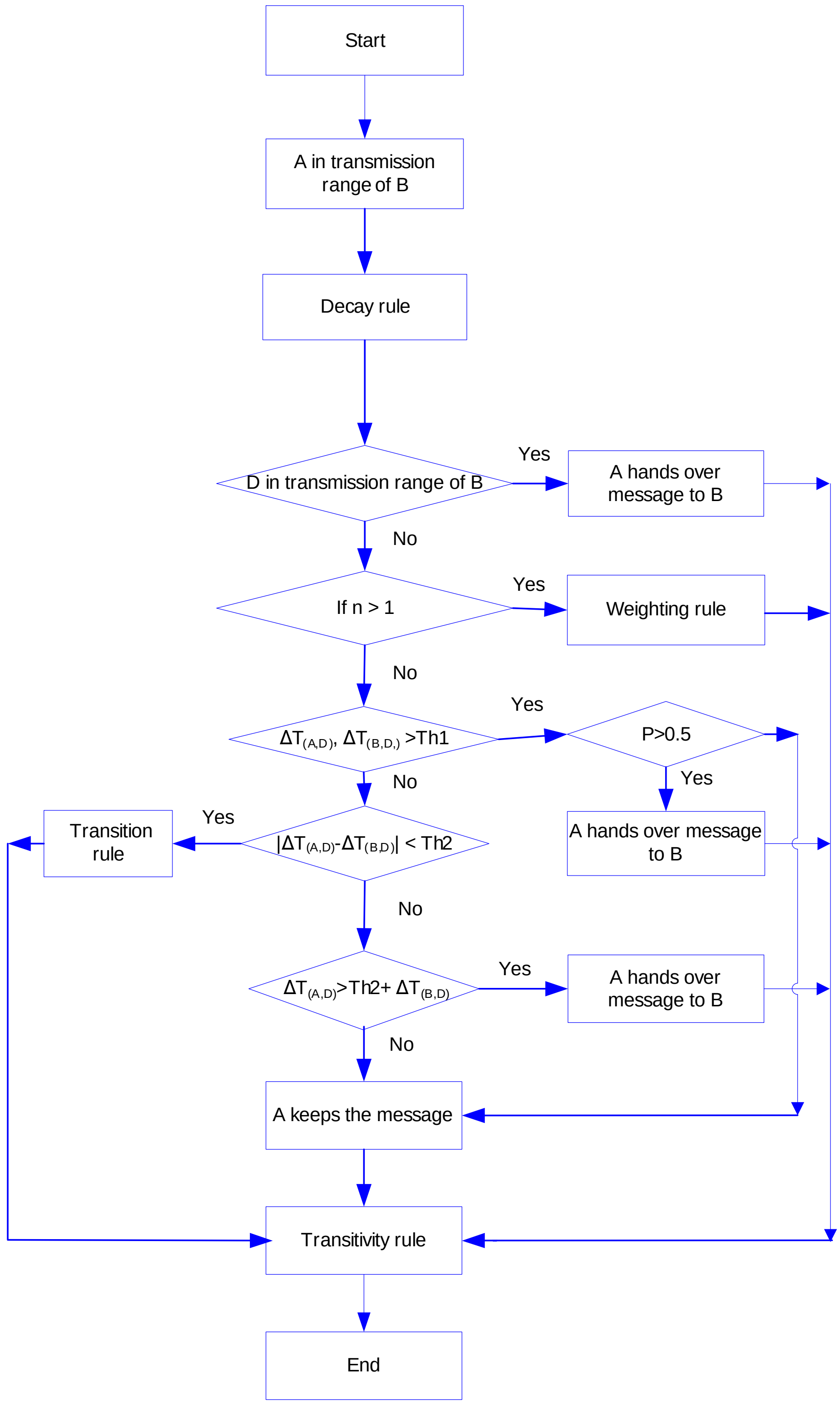


Figure 4.3: The flowchart of the SARP

a large range of node heterogeneity to be captured. The network in this mobility model consists of a number of nodes moving independently on a 2-dimensional torus in a geographical region. Each node transmits up to distance K ≥ 0 meters away, and each message transmission takes one time unit. Euclidean distance is used to measure the proximity between two nodes (or their positions) A and B. A slotted collision avoidance MAC protocol with Clear-to-Send (CTS) and Request-to-Send (RTS), has been implemented in order to arbitrate between nodes contending for the shared channel. The network model used for the proposed protocol varies from being extremely sparse to almost connected networks for a suite of different application scenarios. The performance of the SARP were compared to those of the other protocols with heterogeneous node mobility and mobility showing a strong location preference. In this more realistic mobility scenario, some relays are mobile enough to carry data throughout the network. In this case, not all relays are equally helpful for the delivery process. The proposed protocol can often recognize these nodes, because the nodes that move more often also encounter different destinations more often, and also take advantage of the high locality of many of the nodes.

The nodes are classified into four groups: (1) 40% of the nodes move locally most of the time (pl [0.8, 0.9]) but may occasionally roam into other preferred communities (pr [0.05, 0.15]); (2) 30% of the nodes move only locally inside their own community(pl = 1, pr = 0); (3) 20% of the nodes quite often roam outside their community (pl = [0.2, 0.3] and pr = [0.5, 0.7]); (4) 10% of the nodes are static and uniformly distributed in the network. The proposed technique is compared to the following protocols.

- Epidemic routing (epidemic) [29]
- Optimal (binary) Spray and Wait (SW) [28]
- Spray and Focus (SF) [52]

- Most Mobile First strategy [54]
- Self-Adaptive Routing Protocol (SARP) [63]

For all the protocols, an attempt has been made to tune the parameters in each scenario separately, in order to achieve good transmission-delay. The utility's threshold parameter for Spray and Focus, and SARP is set to 130 and 30, respectively.

4.4.2 Evaluation Scenarios

In the simulation, 110 nodes move according to the community-based mobility model [52] in a 500 x 500 2-dimensional torus in a given geographical region. The message inter-arrival time is uniformly distributed in such a way that the traffic can be varied from low (10 messages per node in 30,000 time units) to high (70 messages per node in 30,000 time units). The message time to live (TTL) is set to 9,000 time units. Each source node selects a random destination node, begins generating messages to it during simulation time.

The performance of the protocols is evaluated with respect to the impact of the number of message copies. Second, with respect to the low transmission range and varying buffer capacity under high traffic load. Third, with respect to the moderate-level of connectivity and varying traffic load. Fourth, the performance of the protocols is examined in terms of the bandwidth. Finally, the performance of the protocols is examined in terms of the level of connectivity changes.

Impact due to Number of Message Copies

We firstly look into impact of the number of message copies toward the performance of each protocol. The transmission range K of each node is set to 30 meters, leading to a relatively sparse network. In order to reduce the effect of contention on any shared channel, the traffic load and buffer capacity is set to medium (i.e., 40 generated messages per node in 30,000 time units) and high (i.e. 1000 messages), respectively. The number of message copies is then increased from 1 to 20 in order to examine their impact on the effectiveness of each protocol. The proposed SARP is compared with the S&F, S&W, and MMF schemes, since each scheme has a predefined L to achieve the best data delivery. Note that the value of L depends on the application requirements, the mobility model considered, and the design of the protocol.

Figure 4.4 shows the results on message delivery delay, delivery ratio, and number of transmissions under different numbers of copies of each generated message. As can be seen, the L value has a significant impact on the performance of each scheme. It is observed that best performance can be achieved under each scheme with a specific value L. These L values can serve as a useful rule of thumb for producing good performance.

Effect of Buffer Size

The performance of SARP is examined under the situation the network is sparse (under low transmission range i.e., K = 30), with high traffic load (60 messages generated per node), and varying buffer space capacity. If a node encounters another node and has limited remaining buffer space, a portion of the messages that should be forwarded can not be delivered even though the encountered node metric is better than the custodian node. This situation results in extra queuing delay, especially in the case of flooding schemes. The performance of the SARP was examined with respect to different buffer space values. Figure 4.5 compares, with different buffer space capacities, the delivery delay, number of transmissions performed, and the delivery ratio produced by the considered protocols. The buffer space was varied from 5 (limited capacity) to 200 (relatively high capacity) messages to reflect the performance of the protocols under the considered traffic load. As shown, when

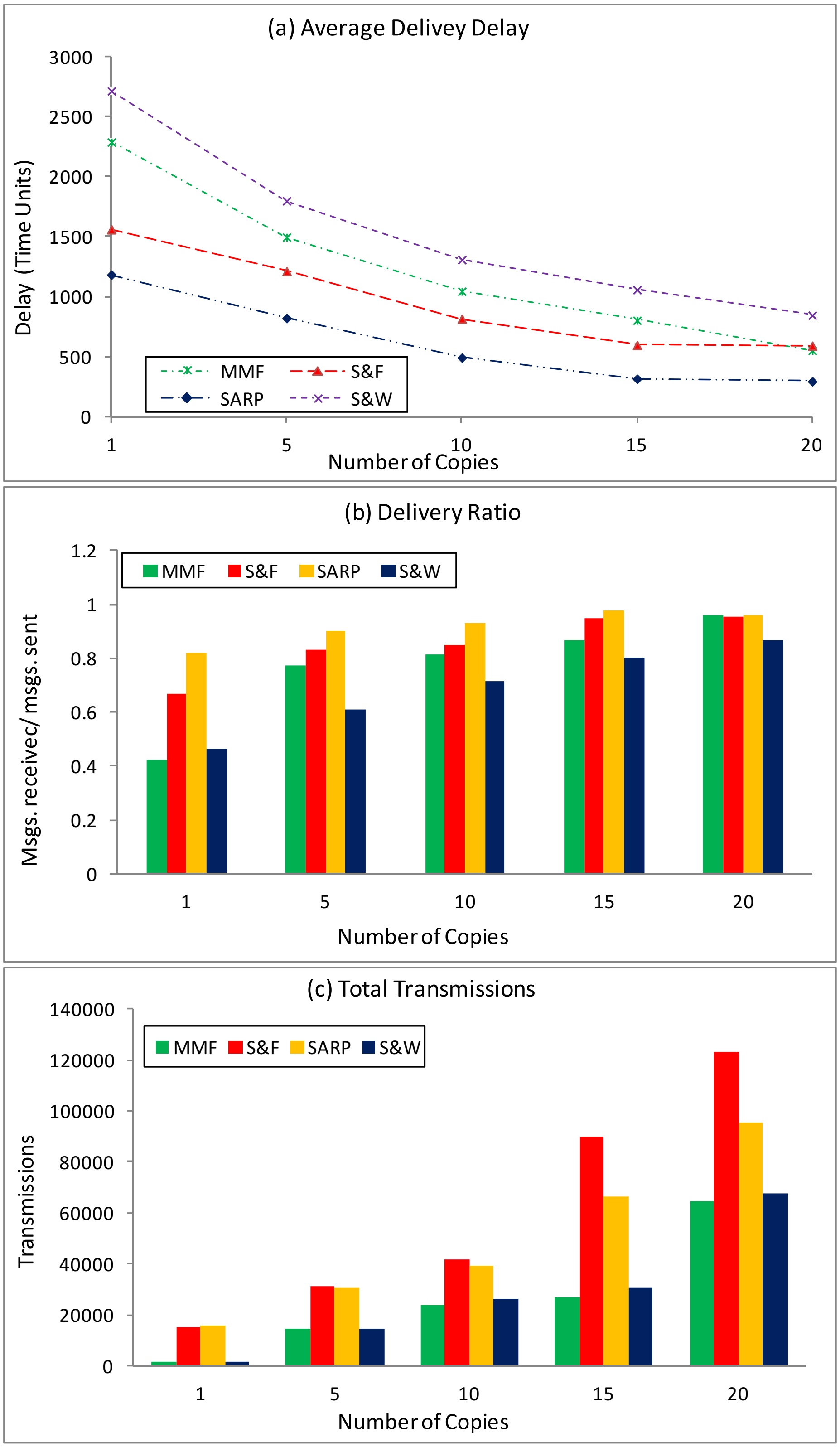


Figure 4.4: Impact of the number of message copies.

the buffer size is small (less than 100 message in our case) the performance of the protocols is highly affected by the capacity of the buffer, especially the Epidemic routing.

Epidemic routing produces the worst delivery delay in all scenarios, since it is affected by both the limited buffer size and the mobility model; SARP produces the lowest delivery delay and highest delivery ratio. SARP yielded a shorter delivery delay than S&W by 67%, and a higher delivery ratio than S&W by 80%. SARP produced delay shorter than S&F by 32%, and higher delivery ratio than S&F by 22%. Although SARP produced more transmissions than MMF, it yielded a smaller delivery delay than that of MMF by 51%. As the buffer size increased, the performance of all protocols was improved especially for MMF. When the buffer size is larger than the traffic demand, the MMF scheme has yielded a competitive performance due to the relaxation of buffer capacity limitation. SARP still yielded the best performance with a smaller number of transmissions than S&F by 17%.

Effect of Traffic Load

This scenario is similar to the previous scenario but with variation in the traffic load. Each node attempts to randomly select a destination node, begins generating messages and continues to increase the rate which results in average traffic loads, i.e., the total number of messages generated throughout the simulation. The main goal of this scenario is to increase contentions on wireless channel and observe how the contentions affect the performance of the protocols. The protocols have been examined under low bandwidth value (i.e., two messages per unit of time), which makes it insufficient to enable some contacts to forward all intended messages. The protocols have been examined for two-buffer sizes: 1) unlimited capacity; and 2) low capacity (15 messages). Figure 4.6 shows the performance of all the routing algorithms in terms of the average delivery delay, delivery ratio, and total number

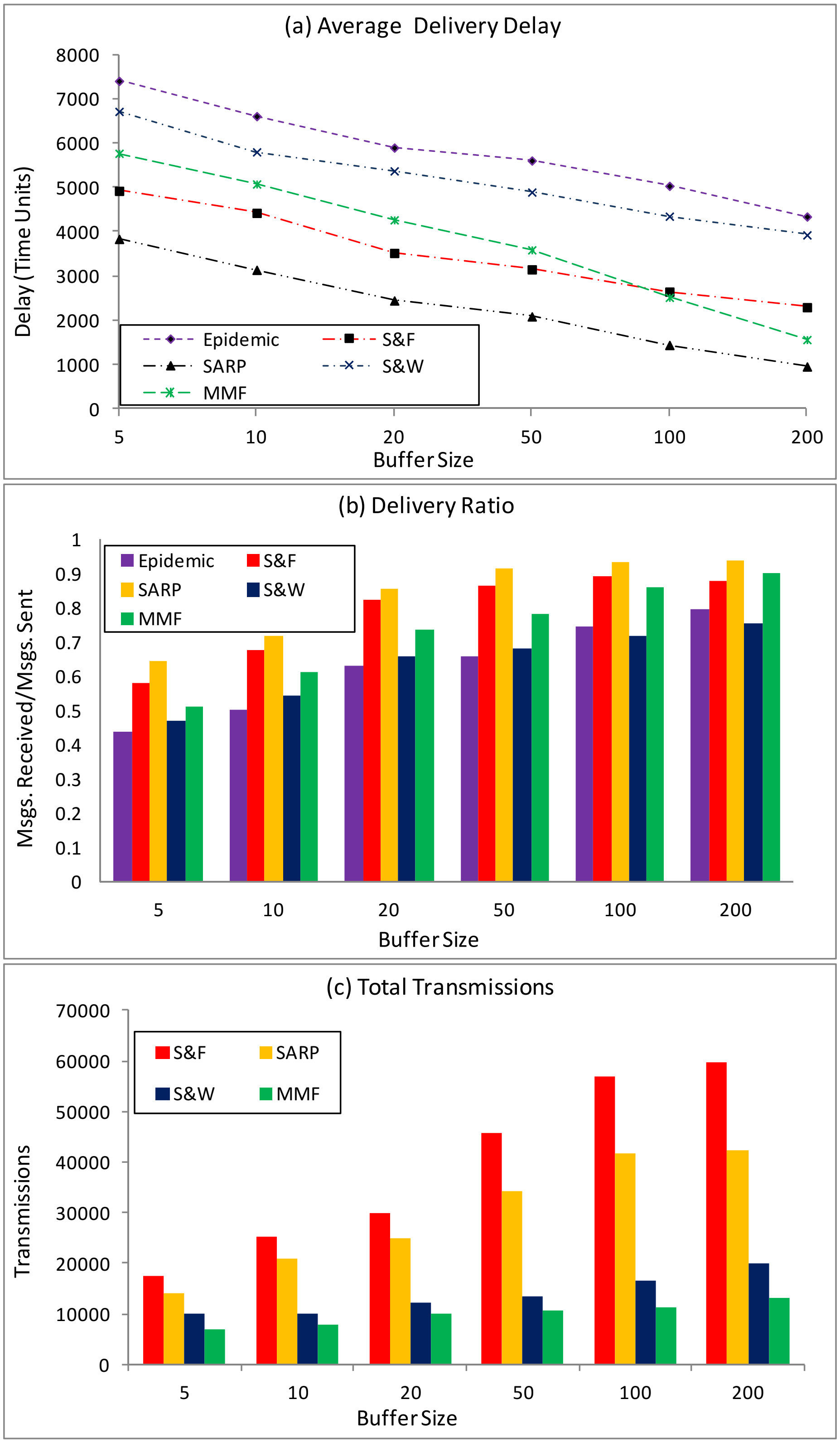


Figure 4.5: The effect of buffer size

of transmissions. In this scenario the buffer size effect is relaxed, and transmission range is set to 80 meters (highly connected network).

Epidemic routing produces the longest delivery delay and requires higher number of transmissions compared to all other schemes. It produces delivery delay at-least four times longer than the SARP does and require number of transmissions at least an order of magnitude higher than the SARP scheme. Thus its not included in the figures. From the results shown in Figure 4.6, we observe that as the traffic load increases, the available bandwidth decreases, as a result the performance of the protocols decreases. Epidemic routing has the lowest performance. When the traffic load is moderate (less that 50 messages), it is clear that the MMF outperforms all existing multiple-copy routing protocols. This is because in MMF, the effect of buffer size is relaxed making all the nodes that are marked as roaming nodes buffer unlimited number of messages while roaming between communities. The SARP scheme shows the second best delivery delay. SARP can produce delay up to 0.97 of that of MMF, faster than that of S&F and S&W by 86% and 280%, respectively. SARP and MMF produce the best delivery ratio compared to all existing schemes. SARP and MMF can achieve delivery ratio above 96% and 97% respectively, while the epidemic routing degrades below 50% for high traffic loads. S&F protocols can achieve delivery ratio above 92%.

As the traffic load exceeds 50 messages per node, the contention on wireless channel become higher. The performance of all schemes is decreased since the busy links cause long delays. The delivery delay obtained by the SARP is faster 26% than that of MMF.

As the buffer capacity reduced to low capacity (e.g. 10 messages), and the traffic load increases; the available bandwidth decreases and the buffer occupancy increases. When the traffic increases, it is clear that the proposed approach outperforms all existing multiple-copy routing protocols. The SARP scheme obtains the

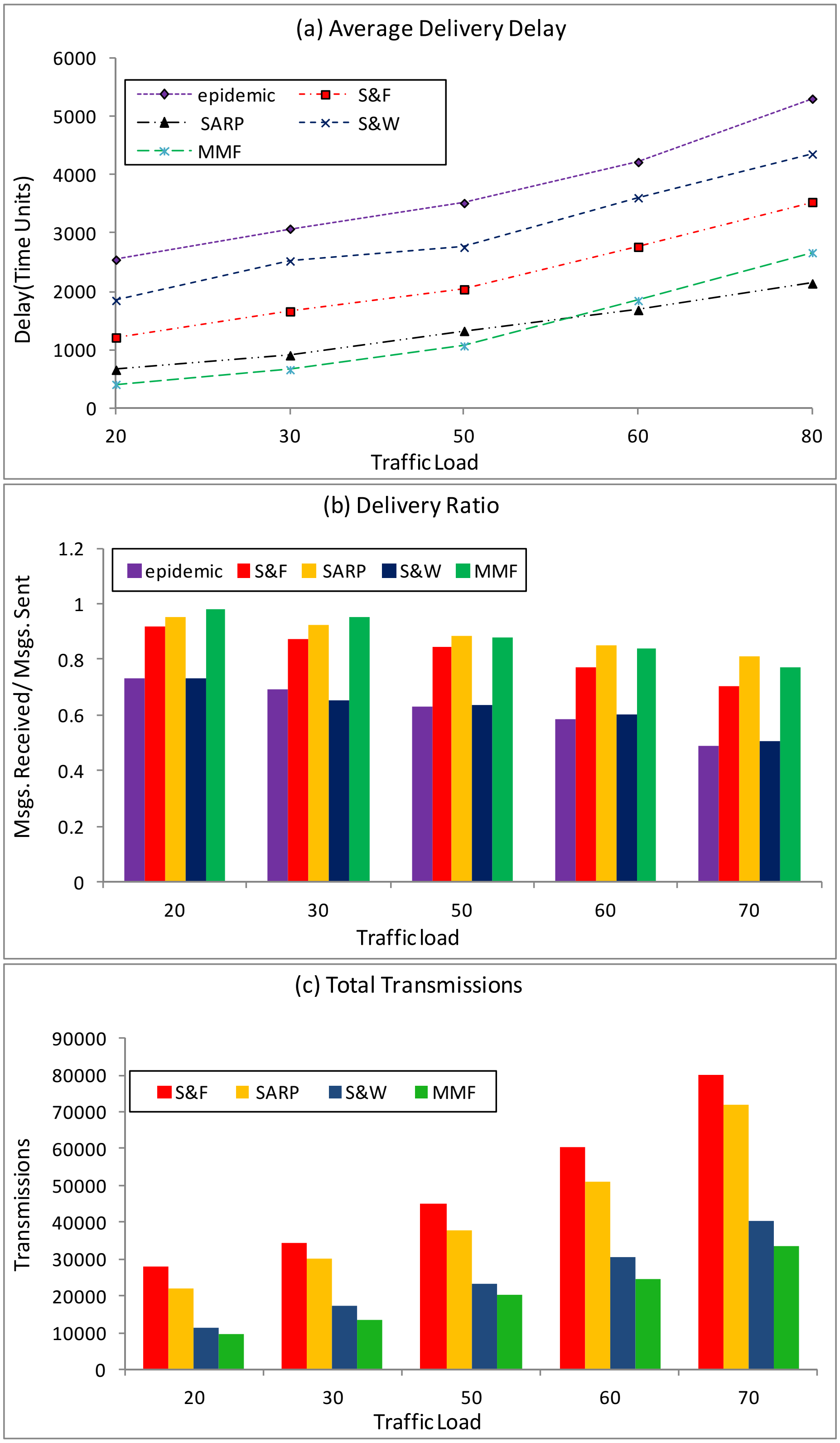


Figure 4.6: The effect of traffic load under unlimited buffer capacity.

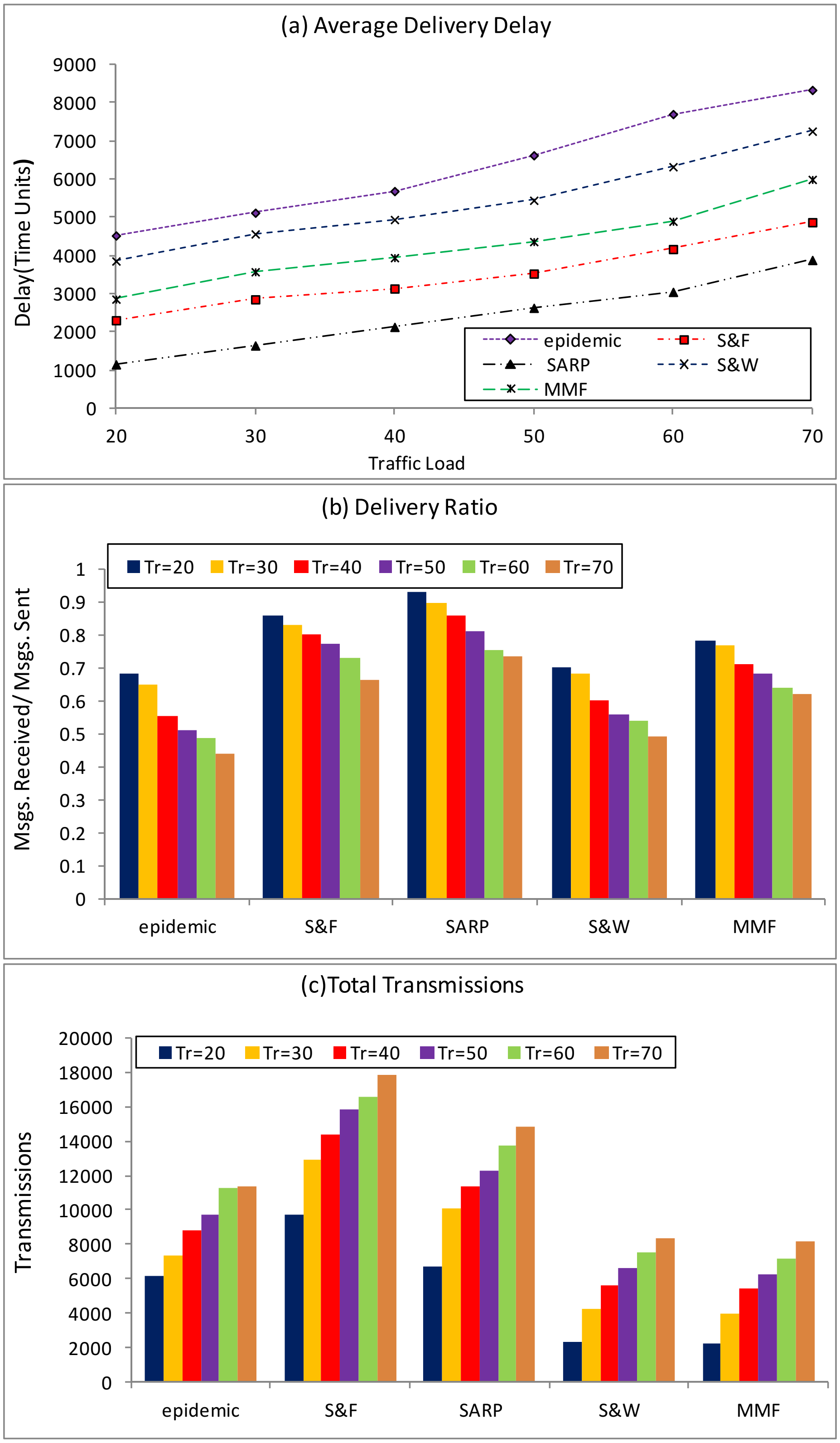


Figure 4.7: The effect of traffic load under limited buffer capacity.

fastest delivery delay and the best delivery ratio compared to all existing schemes. It can be faster than MMF by 58%, S&W by 93%, and S&F by 31%. Although the SARP scheme requires more transmissions compared to the MMF, the number is still smaller than that produced by S&F. For high traffic loads (70 generated messages per node), SARP can achieve delivery ratio above 73% , while the epidemic routing degrades below 45%, S&F below 66%, MMF and S&W below 60%. When the traffic load is low (below 50 messages), SARP outperforms all other schemes in terms of delivery delay. It is faster than Epidemic by 310%, S&F by 96%, MMF by 245%, and S&W by 280%. That's because the contention on wireless channel is low, making SARP employing its routing policy more efficiently. Figure 4.7 shows the performance of all techniques under this scenario.

The Effect of Connectivity

This scenario studies the performance of the protocols at different connectivity ranges. The level of connectivity ranges from very sparse to highly connected networks by varying the value of K. This scenario examines the performance of the SARP in the cases when the network is highly congested and the demand on the wireless channel is very high. The buffer capacity is kept low (15 messages), and the traffic load is considerably high (60 messages). Figure 4.8 shows the average delay and the number of transmissions as a function of transmission range.

From the results shown in the figures, a number of interesting observations can be made about these figures. First, although Epidemic routing performs too many transmissions, it is still far from achieving competitive delays because of the contention caused by increasing the demand on the wireless channel. Second, the SARP scheme outperforms all protocols in terms of delivery delay with fewer transmissions than the S& F scheme, for either low levels or high levels of connectivity. When the network is sparsely connected, the performance of all schemes is affected

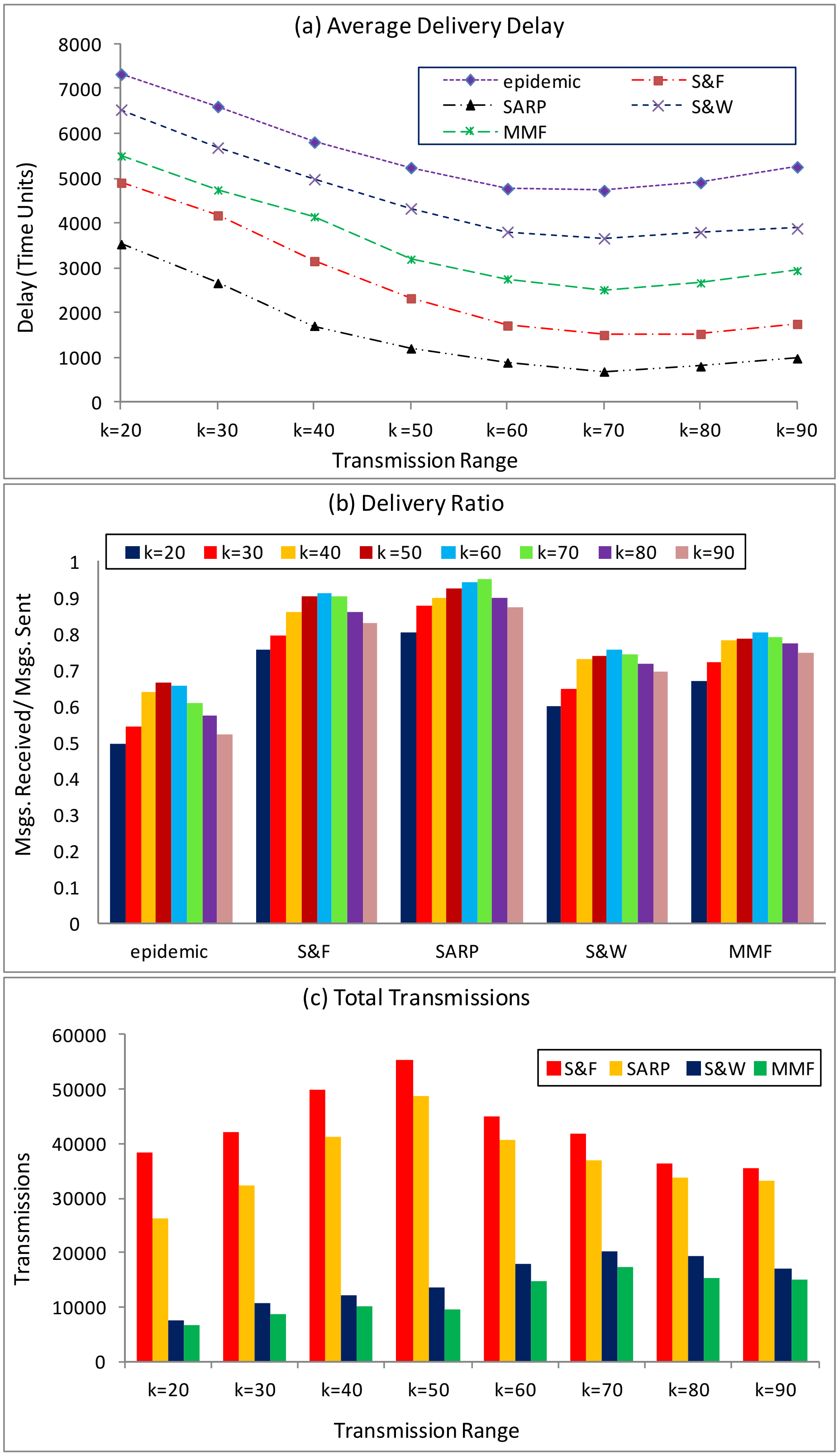


Figure 4.8: The effect of connectivity.

by the uncertainty of buffer occupancy status. On the other hand, when the network is moderate-connected, the SARP scheme can achieve the best performance of delivery delay compared to all other schemes with more transmissions. As the network becomes almost connected and the traffic load is high, the uncertainty of both buffer occupancy status and the availability of bandwidth affect the performance of all the techniques. the SARP still outperforms all other schemes in terms of delivery delay and delivery ratio

4.5 Summary

This chapter has described the framework of the proposed encounter-based routing protocol for DTNs: the SARP. The system model and the mobility model have been described, and the proposed protocol introduced, along with its forwarding and decision strategies. The proposed protocol has been implemented and tested by means of simulations. To examine its effectiveness, the protocol has been compared to four existing routing protocols. The results show that the proposed protocol outperforms the other protocols for the considered mobility scenarios.

Chapter 5

Contention Aware Self Adaptive Routing Protocols for DTNs

5.1 Introduction

Although SARP has improved the previously reported designs in terms of the message delivery delay delivery ratio, and the number of transmissions, it is still subject to respective problems and implementation difficulties. It suffers from contention in case of high traffic loads, in which packet drops could result in a significant degradation of performance and scalability. None of previously reported spray routing schemes have fully investigated how the protocol should take advantage of dynamic network status to improve the performance, such as packet collisions, wireless link conditions, and nodal buffer occupancy. More importantly, the channel capacity and buffer occupancy states have never been jointly considered in the derivation of utility functions. These two factors could be overlooked/ignored if the encounter frequency is low, where the routing protocol performance is dominated by node mobility, while the network resource availability does not plays an important role. However, in the scenario that the nodal encounter frequency is large and each node has many choices for packet forwarding, the network resource availability could become a critical factor for improving routing protocol performance, and should be taken seriously in the derivation of utility functions.

With this in mind, we introduce two novel DTN routing protocols, called Self Adaptive Utility-based Routing Protocol (SAURP), and Adaptive Reinforcement-Based Routing Protocol (ARBRP) that overcome the shortcomings of the previously reported multi-copy schemes. Each of the schemes employs the dynamic network status in a different way. SAURP employing the network dynamic in a form of quality metric function in a form of enter-encounter time between nods, while ARBRP employs it in form of contact time durations between nods.

5.2 Self Adaptive Utility-based Routing Protocol (SAURP)

The main feature of SAURP is its ability in adaptation to the fluctuation of network status, traffic patterns/characteristics, and user behaviors, so as to reduce the number of transmissions, message delivery time, and increase delivery ratio. This is achieved by jointly considering node mobility statistics, congestion, and buffer occupancy, which are subsequently fused in a novel quality-metric function. In specific, the link availability and buffer occupancy statistics are obtained by sampling the channels and buffer space during each contact with another node. We use time-window based update strategy because it is simple in implementation and rather robust against parameter fluctuation. Note that the network conditions could change very fast and make a completely event-driven model unstable. The developed quality-metric function targets to facilitating decision making for each active data message, resulting in optimized network performance. Figure 5.1 illustrates the functional modules of the SAURP architecture along with their relations.

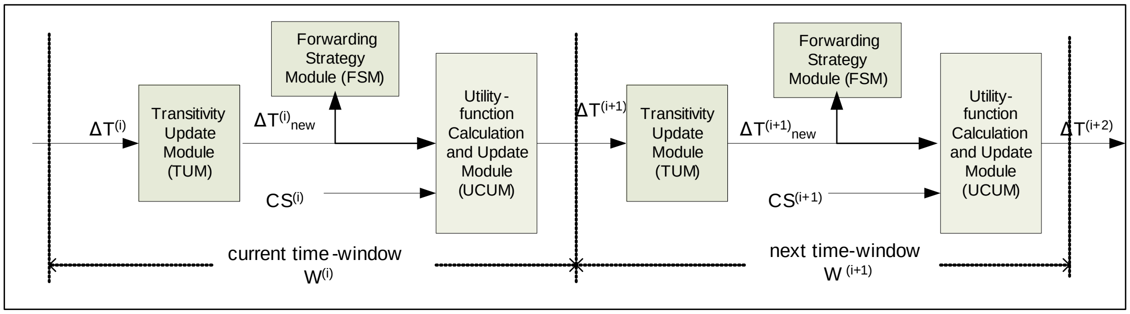


Figure 5.1: The SAURP Architecture

The Contact Statistics (denoted as $CS^{(i)}$) is obtained at each node regarding the total nodal contacts durations, channel condition, and buffer occupancy state. These values are collected at the end of each time window and used as one of the two inputs to the Utility-function Calculation and Update Module (UCUM). Another input to the UCUM, as shown in Figure5.1, is the updated utility denoted as $\triangle T_{new}^{(i)}$, which is obtained by feeding $\triangle T^{(i)}$ through the Transitivity Update Module (TUM). UCUM is applied such that an adaptive and smooth transfer between two consecutive time windows (from current time-window to next time-window) is maintained. $\triangle T^{(i+1)}$ is the output of UCUM, and is calculated at the end of current time window $W^{(i)}$. $\triangle T^{(i+1)}$ is thus used in time window $W^{(i+1)}$ for the completely the same tasks as in window $W^{(i)}$.

Forwarding Strategy Module (FSM) is applied at the custodian node as a decision making process when encountering any other node within the current time window based on the utility value (i.e., $\triangle T^{(i)}$).

It is important to note that CS, TUM, FSM, and message vector exchange are event-driven and performed during each contact, while UCUM is performed at the end of each time-window. The following subsections introduce each functional module in detail.

5.2.1 Contact Statistics (CS)

To compromise between the network state adaptability and computation complexity, each node continuously updates the network status within a fixed time window. The maintained network states are referred to as Contact Statistics (CS), which include nodal contact durations, channel conditions, and buffer occupancy state, and will be fed into UCUM at the end of each time window. The CS collection process is described as follows.

Let two nodes A and B are in the transmission range of each other, and each broadcasts a pilot signal per k time units in order to look for its neighbors within its transmission range. Let $T_{(A,B)}$, T_{free}, and T_{busy} represent the total contact time, the amount of time the channel is free and the buffer is not full, and the amount of time the channel is busy or the buffer is full, respectively, at node A or B during time window $W^{(i)}$. Thus, the total duration of time in which node A and B can exchange information is calculated as:

$$T_{free} = T_{(A,B)} - T_{busy} \tag{5.1}$$

Note that the total contact time could be accumulated over multiple contacts between A and B during $W^{(i)}$.

5.2.2 Utility-function Calculation and Update Module (UCUM)

UCUM is applied at the end of each time window and is used to calculate the currently observed utility that will be further used in the next time window. The two inputs to UCUM in time window $W^{(i)}$ are: (i) the predicted inter-contact time ($\triangle T^{(i)}$), which is calculated according to the previous time-window utility (i.e., $\triangle T^{(i)}$), as well as an update process via the transitivity property update (introduced in subsection 3.3), and (ii) the observed inter-contact time obtained from the current $CS^{(i)}$ (denoted as $\triangle T_{cs}^{(i)}$).

Calculation of inter-contact Time ($\triangle T^{(i)}$)

An eligible contact of two nodes occurs if the duration of the contact can support a complete transfer of at least a single message between the two nodes. Thus, in the event that node A encounters B for a total time duration T_{free} during time window $W^{(i)}$, the number of eligible contacts in the time window is determined by:

$$n_c = \left\lfloor \frac{T_{free}}{T_p} \right\rfloor \tag{5.2}$$

where T_p is the least time duration required to transmit a single message. Let $\triangle T_{cs(A,B)}^{(i)}$ denotes the average inter-contact time duration of node A and B in time $W^{(i)}$. Obviously, $\triangle T_{(A,B)}^{(i)} = \triangle T_{(B,A)}^{(i)}$. We have the following expression for$\triangle T_{cs(A,B)}^{(i)}$:

$$\triangle T_{cs(A,B)}^{(i)} = \frac{W^{(i)}}{n_c^{(i)}} \tag{5.3}$$

$\triangle T^{(i)}{}_{cs(A,B)}$ describes how often the two nodes encounter each other per unit of time (or, the encounter frequency) during time $W^{(i)}$ considering the event the channel is busy or the buffer is full.

Thus, inter-contact time of a node pair intrinsically relies rather on the duration and frequency of previous contacts of the two nodes than simply on the number of previous contacts or contact duration. Including the total duration of all the contacts (excluding the case when the channel is busy or the buffer is full) as the parameter is expected to better reflect the likelihood that nodes will meet with each other for effective message exchange. With this, the proposed routing protocol does not presume any knowledge of future events, such as node velocity, node movement direction, instants of time with power on or off, etc; instead, each node keeps network statistic histories with respect to the inter-contact frequency of each node pair (or, how often the two nodes encounter each other and are able to perform an effective message exchange).

Time-window Transfer Update

Another important function provided in UCUM is for the smooth transfer of the parameters between consecutive time windows. As discussed earlier, the connectivity between any two nodes is measured according to the amount of inter-contact time during $W^{(i)}$, which is mainly based on the number of contacts (i.e., n_c) and the contact time (i.e., T_{free}). These contacts and contact durations may change dramatically from one time window to the other and address significant impacts on the protocol message forwarding decision. Hence, our scheme determines the next time window parameter using two parts: one is the current time window observed statistics (i.e., $\triangle T_{cs}^{(i)}$), and the other is from the previous time window parameters (i.e., $\triangle T^{(i)}$), in order to achieve a smooth transfer of parameter evolution. The following equation shows the derivation of $\triangle T^{(i+1)}$ in our scheme.

$$\triangle T^{(i+1)} = \gamma . \triangle T_{cs}^{(i)} + (1 - \gamma) \triangle T^{(i)} \tag{5.4}$$

The parameter γ is given by

$$\gamma = \frac{\mid \triangle T^{(i)} - \triangle T_{cs}^{(i)} \mid}{max(\triangle T^{(i)}, \triangle T_{cs}^{(i)})}, \; \triangle T^{(i)}, \; \triangle T_{cs}^{(i)} > 0 \tag{5.5}$$

If $\triangle T_{cs}^{(i)} > W$, which happens if $n_c^{(i)} = 0$, then $\triangle T^{(i+1)} = \frac{2W}{n_c^{(i-1)}}$. This case represents a worst case scenario, i.e., unstable node behavior, or low quality of node mobility. Hence, the $\triangle T^{(i+1)}$ value should be low.

$\triangle T^{(i+1)}$ represents the routing metric (utility) value that is used as input to the next time window. This value is maintained as a vector of inter-encounter time that is specific to every other node, which is employed in the decision making process for message forwarding.

5.2.3 Transitivity Update Module (TUM)

When two nodes are within transmission range of each other, they exchange utility vectors regarding the message destination. With the update, the custodian node decides whether or not the message should be forwarded to the encountered node. This exchange of summary vectors is followed by another update, called transitivity update. Although the idea of using transitivity and time-window updates are not new [43, 67], the proposed SAURP has gone through a much different way. The transitivity property [43] based on the observation that if node A frequently encounters node B and B frequently encounters node D, then A has good ability to forward messages to D through B. We formulated the updating rule as follows:

$$\triangle T^{(i)}_{(A,D)new} = \alpha \triangle T^{(i)}_{(A,D)} + (1-\alpha)(\triangle T^{(i)}_{(A,B)} + \triangle T^{(i)}_{(B,D)}) \tag{5.6}$$

where w is weighting factor that must be less than 1 to be valid.

$$\alpha = \frac{\triangle T^{(i)}_{(A,B)} + \triangle T^{(i)}_{(B,D)}}{\triangle T^{(i)}_{(A,D)}},\ \triangle T^{(i)}_{(A,D)} > \triangle T^{(i)}_{(A,B)} + \triangle T^{(i)}_{(B,D)} \tag{5.7}$$

α has a significant impact on the routing decision rule. From theoretical perspective, when a node is encountered that has more information for a destination, this transitivity effect should successfully capture the amount of uncertainty to be resolved regarding the position of the destination. Thus, a transitivity property is needed to update values only when $\triangle T^{(i)}_{(A,D)} > \triangle T^{(i)}_{(B,D)}$ in order to ensure that node

A reaches D through B. Otherwise, if $\triangle T^{(i)}_{(A,D)} < \triangle T^{(i)}_{(B,D)}$, the transitivity property is not useful since node A is a better candidate for forwarding messages directly to node D rather than forwarding them through B. This rule is applied after nodes finish exchange messages.

5.2.4 The Forwarding Strategy Module (FSM)

The decision of message forwarding in SAURP is mainly based on the goodness (utility function value) of the encountered node regarding the destination, and the number of message copy tokens. If the message tokens are greater than 1, weighted copy rule is applied, the forwarding rule is applied otherwise.

The Weighted Copy Rule

The source of a message initially starts with L copies; any node A that has $N_A > 1$ message copy tokens (source or relay) and that encounters another node B with no copies and $\triangle T^{(i)}_{(B,D)} < \triangle T^{(i)}_{(A,D)}$, node A hands over to node B a number of copies according to its goodness for the destination node D. Node A hands over some of the message copy tokens to node B and keeps the rest for itself according to the following formula:

$$N_B = \left\lfloor N_A \left(\frac{\triangle T^{(i)}_{(A,D)}}{\triangle T^{(i)}_{(B,D)} + \triangle T^{(i)}_{(A,D)})} \right) \right\rfloor \tag{5.8}$$

where N_A is the number of message tokens that node A has, $\triangle T^{(i)}_{(B,D)}$ is the inter-contact time between node B and node D, and $\triangle T^{(i)}_{(A,D)}$ is the inter-contact time between nodes A and D. This formula guarantees that the largest number of message copies is spread to relay nodes that have better information about destination node. After L messages have been copied to custodian nodes, each of the L nodes carrying a copy of the message performs according to the forwarding rule as descried next.

Algorithm 5.1 The forwarding strategy of SAURP

On contact between node A and B
Exchange summary vectors
for every message M at buffer of custodian node A *do*
    if destination node D in transmission range of B *then*
        A forwards message copy to B
    end if
    else if $\triangle T^{(i)}_{(A,D)} > \triangle T^{(i)}_{(B,D)}$ *do*
        if the message tokens >1 *then*
            apply weighted copy rule
        end if
        else if $\triangle T^{(i)}_{(A,D)} > \triangle T^{(i)}_{(B,D)} + T_{th}$ *then*
            A forwards message to B
        end else if
    end else if
end for

The Forwarding Rule

- If the destination node is one hop away from an encountered node, the custodian node hands over the message to the encountered node.
- If the inter-contact time value of the encountered node relative to that of the destination node is less than that of the custodian node by a threshold value, $\triangle T_{th}$, a custodian node hands over the message to the encountered node.

The complete mechanism of the forwarding strategy in SAURP is summarized as shown in Algorithm 5.1.

5.2.5 Statistical Study

In this section a statistical analysis is conducted on the performance of the proposed SUARP. Without loss of generality, Community-Based Mobility Model [52] is employed in the analysis. The problem setup consists of an ad hoc network with a number of nodes moving independently on a 2-dimensional torus in a geographical region, and each node belongs to a predetermined community. Each node can transmit up to a distance $K \geq 0$ meters away, and each message transmission takes one time unit. Euclidean distance is used to measure the proximity between two nodes (or their positions) A and B. A slotted collision avoidance MAC protocol with Clear-to-Send (CTS) and Request-to-Send (RTS), is implemented for contention resolution. A message is acknowledged if it is received successfully at the encountered node by sending back a small acknowledgment packet to the sender.

The performance measures in the analysis include the average delivery probability and the message delivery delay. The analysis is based on the following assumptions.

- Nodes mobility is independent and heterogeneous, where nodes have frequent appearance in some locations.

- Each node in the network maintains at least one forwarding path to every other node. Figure 5.2 illustrates the paths that a message copy may take to reach the destination.

- Each node belongs to a single community at a time (representing some hot spots such as classrooms, office buildings, coffee shops), and the residing time on a community is proportional to its physical size.

- The inter-contact time $\triangle T_{(A,B)}$ between nodes A and B follows an exponential distribution with probability density function (PDF), $P_{\triangle T_{(A,B)}}(t) = \beta_{(A,B)}.e^{-\beta_{(A,B)}t}$, where t is the time instance.

It has been shown that a number of popular mobility models have such exponential tails (e.g., Random Walk, Random Waypoint, Random Direction, Community-based Mobility [50, 5]). In practice, recent studies based on traces collected from real-life mobility examples argued that the inter-contact time and the contact durations of these traces demonstrate exponential tails after a specific cutoff point [66].

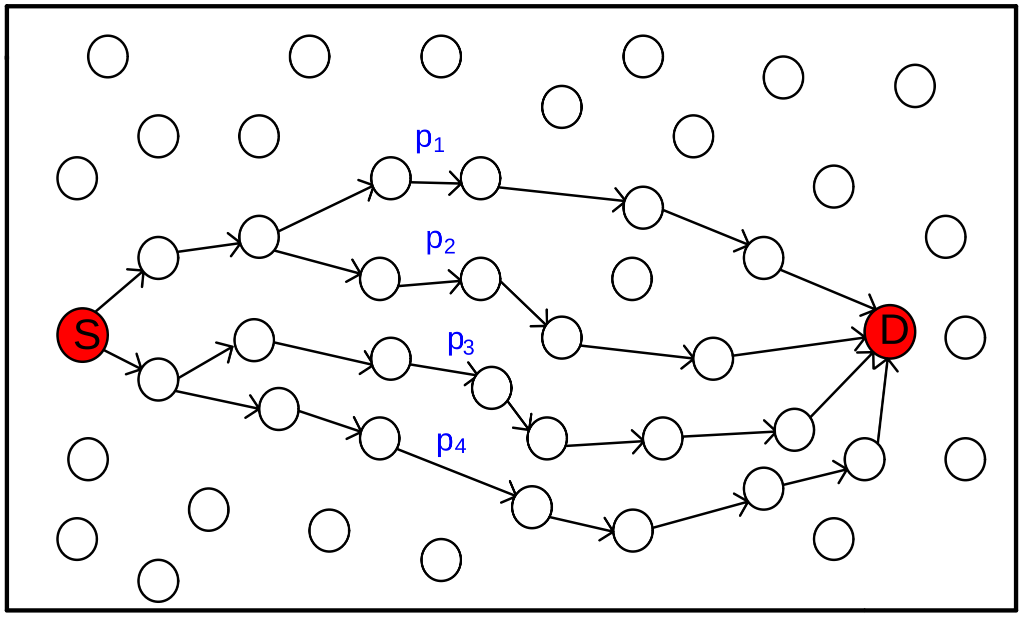


Figure 5.2: Paths of message copies to destination

Based on the mobility model of the nodes, the distribution of the inter-contact time can be predicted and calculated using time widow updates shown in (5.4). Thus, parameter β_{AB} is calculated as $\beta_{AB} = \frac{1}{\triangle T_{(A,B)}}$.

Delivery Probability

In order to calculate the expected message delivery ratio, any path of message m between S and D is a $k - hop$ simple path, denoted as l, which is represented by a set of nodes and links denoted as $\{S, h_1 \, h_2 \, ...h_{k-1}, D\}$, and $\{e_1, e_2, ..., e_k\}$, respectively. The cost on each edge, denoted as $\{\beta_1, \beta_2, .., \beta_k\}$, is the inter-contact rate (or frequency) of each adjacent node pair along the path. According to the forwarding policy of SAURP, the values of inter-contact rate should satisfy $\{\beta_1 < \beta_2 < .. < \beta_k\}$. The path cost, $PR_l(t)$, is the probability that a message m is successfully forwarded from S to D along path l within time t, which represents a cumulative distribution function (CDF). The probability density function of a path l with $k - hop$ for one message copy can be calculated as convolution of k probability distributions [73] which is calculated as:

$$Pr_l(t) = p_1(t) \otimes p_2(t) \otimes ...p_k(t) \tag{5.9}$$

Let the probability density function (PDF) for the message delivery along a one-hop path i be denoted as $p_i(t) = \beta_i e^{-\beta_i t}$. Thus, the PDF for a $k-hop$ simple path l with an edge cost $\{\beta 1,\ \beta 2,\ ...,\beta_k\}$ can be expressed as

$$Pr_l(t) = \sum_{i=1}^{k_l} C_i^{(k_l)} p_i(t) \tag{5.10}$$

where the coefficients are given as follows:

$$C_i^{(k_l)} = \prod_{j=1, i \neq j}^{k_l} \frac{\beta_j}{\beta_j - \beta_i} \tag{5.11}$$

The proof is provided in Appendix.

The probability of message delivery on forwarding path l between any source S, and destination D, within expiration time T is expressed as :

$F_l(T) = PR_l(Td_l < T) = \int_0^T Pr_l(t)dt$
$= \sum_{i=1}^{k_l} C_i^{k_l} \int_0^T P_i(t)dt$

$$PR_l(Td_l < T) = \sum_{i=1}^{k_l} C_i^{(k_l)}.(1 - e^{-\beta_i T}) \tag{5.12}$$

If there are $L-1$ copies (excluding the message at the source) of message m traversing through $L-1$ independent paths in the network, the maximum probability of message delivery can be written as

$$PR_{max}(T_d < T) = max\{PR_{SD,} PR_1,\ PR_2, ..PR_{L-1}\} \tag{5.13}$$

where PR_{SD} and PR_l are a random variables represent the delivery probability in case of direct message delivery between S and D, and through one of $L-1$ paths, respectively. The expected delivery probability of message m with $L-1$ copies traversing on $L-1$ paths is calculated as:

$$PR(T_d < T)\ = 1 - PR_{SD}\left(T_{SD} > T\right)\prod_{l=1}^{L-1}\left(1 - PR_l(Td_l < T)\right) \tag{5.14}$$

$$PR(T_d < T)\ = 1 - e^{-\beta_{SD}T}\prod_{l=1}^{L-1}\left(\sum_{i=1}^{k_l} C_i^{(k_l)}\left(e^{-\beta_i T}\right)\right) \tag{5.15}$$

By assuming X totally generated messages in the network, the average of the delivery probability in the network is calculated as

$$PR = \frac{1}{X}\sum_{m=1}^{X} PR_m \tag{5.16}$$

Delivery Delay

The expected total time required to deliver a message from S to D along an individual path l can be calculated as

$E[D_l] = \int_0^\infty PR_l(Td_l > t) = \sum_{i=1}^{k_l} C_i^{(k_l)}.\int_0^\infty e^{-\beta_i t}dt = \sum_{i=1}^{k_l} C_i^{(k_l)}.\frac{1}{\beta_i}$

Let message m have $L-1$ copies (excluding the message at the source) traversing on $L-1$ independent paths. The minimum delivery delay can be written as:

$$D_{SD} = min\{T_{SD,} Td_1,\, Td_2, ..Td_{L-1}\} \tag{5.17}$$

where T_{SD} and Td_l are a random variables representing the delivery delay through direct path between S and D and through one of $L-1$ paths, respectively. The expected delay of message m, $E[D_{SD}]$, can be calculated as

$$E[D_{SD}] = \int_0^\infty P(T_d > t) = \int_0^\infty e^{-\beta_{SD}t} \prod_{l=1}^{L-1} \left(\sum_{i=1}^{k_l} C_i^{(k_l)}.e^{-\beta_i t} \right) dt$$
$$= \tfrac{1}{\beta_{SD}} \int_0^\infty \beta_{SD} e^{-\beta_{SD}t} \prod_{l=1}^{L-1} \left(\sum_{i=1}^{k_l} C_i^{(k_l)}.e^{-\beta_i t} \right) dt =$$
$$\frac{1}{\beta_{SD}} E \left\{ \prod_{l=1}^{L-1} \left(\sum_{i=1}^{k_l} C_i^{(k_l)}.e^{-\beta_i T_{SD}} \right) \right\},\, T_{SD} < \infty \tag{5.18}$$

The above relation gives an upper bound of the delivery delay since it is conditioned to T_{SD}, $T_{SD} < \infty$ and can be taken as point of reference.

The average delivery delay of message m can be calculated intuitively as:

$$E\left[ED_{(S,D)}\right] = \left[\frac{1}{L} \left(T_{SD} + \sum_{l=1}^{L-1} Td_l \right) \right] . \frac{1}{PR(T_d < T)} \tag{5.19}$$

T_{SD} is included in (5.19) only if $T_{SD} < \infty$.

By assuming X totally generated messages in the network, the average delivery delay can thus be calculated as

$$DR = \frac{1}{X} \sum_{m=1}^{X} D_m \tag{5.20}$$

5.2.6 Validation of the Analytical Model

In this section we first validate our analytical model regarding the delivery ratio and delivery delay. Then, the performance of SAURP is further examined via extensive simulations on a number of different scenarios.

Validation of Analytical Model

In order to evaluate the accuracy of the mathematical expressions in this analysis, SAURP is examined under two network status scenarios. In the first scenario, the network is operating under no congestion, i.e., all the nodes have infinite buffer space, and the bandwidth is much larger than the amount of data to be exchanged between any two encountered nodes. In the second scenario, the network is operating under limited resources, i.e., the forwarding opportunities can be lost due to high traffic, limited bandwidth, limited buffer space, or contention (more than one nodes with in range are trying to access the wireless channel at the same time). For both scenarios, 50 nodes move according to community-based mobility model [50] in a 300x300 network size. The transmission range is set to 30 to enable moderate network connectivity with respect to the considered network size. The traffic load is varying from a low traffic load (i.e., 20 messages generated per node in 40,000 time units) to high traffic load (i.e., 80 messages generated per node in 40,000 time units). A source node picks randomly chosen destination and generates messages to it during the simulation time. In this analysis the message copies are set to 5 (i.e., forming a maximum of 5 paths).

Examining SAURP under the two scenarios is very important; in case of no congestion, the best path that is taken by a message is mainly based on the inter-contact time, while under congestion, the message will be buffered for longer period of time and enforced to take longer path to go around the congested area resulting in more dropping rate and longer delivery delay.

To enable accurate analysis, the simulation program is run for a period of time (warm up period of 10,000 time units) such that each node can build and maintain the best forwarding paths with every other node in the network. These forwarding

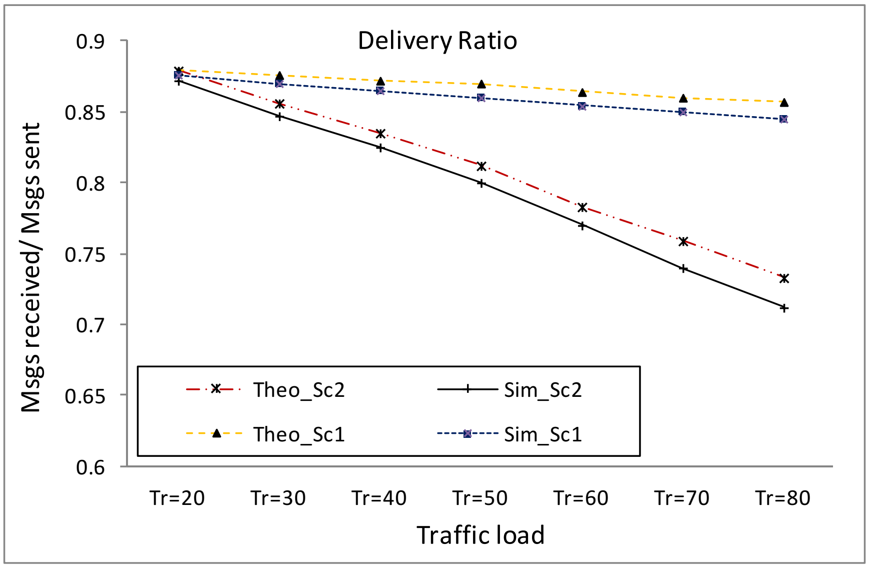


Figure 5.3: The theoretical and simulation results of delivery ratio.

paths are mainly based on the congestion degree (traffic loads values) considered in the analysis. The forwarding path is cached by follow the trajectories of the generated messages during the warm up stage between every source destination pair in the network. These messages are forwarded from node to node according to SAURP routing mechanism.

In this analysis, we simplified the calculation by limiting our study to only the best two of forwarding paths among all other paths and compare the simulation and theoretical results of delivery ratio and delivery delay. In most cases, a message takes the best forwarding path that based on the inter-contacts history if the network is not congested and the buffers operate under their capacity limit.

Figure 5.3 and Figure 5.4, compare theoretical and simulation results of the delivery ratio and delivery delay of the considered scenarios.

As seen from Figure 5.3 and Figure 5.4, when the network resources are enough to handle all the traffic loads (Scenario 1), there is no dramatic change in the obtained delivery ratio and delivery delay for all traffic loads. That is because messages follow the best forwarding paths that lead to best performance. The

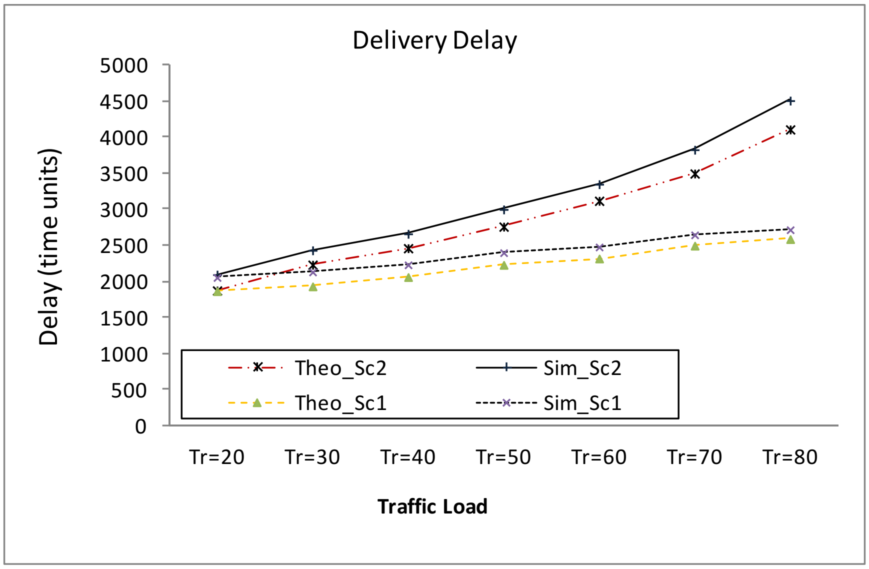


Figure 5.4: The theoretical and simulation results of delivery delay.

simulation and analytical plots for SAURP present close match and validates the generality of the analytical expressions. Additionally, it is evident that (5.15) and (5.19) are tight for all degrees of traffic loads. When the network resources are limited (i.e, scenario 2), the contention and the overhead of MAC layer increase, resulting in longer forwarding paths, higher drop rate, and longer delivery delay. The simulation and analytical plots are still providing close match with small diverge in case of high traffic loads.

Although the contention does affect the accuracy of our theoretical expressions, the error introduced for SAURP is not large (20%), even for large traffic loads. Therefore, we believe the analytical expression is useful in assessing the performance in more realistic scenarios with contention. As an evident by these plots, the actual delay obtained by SAURP becomes increasingly worse than what the theory predicts. This demonstrates the need to add an appropriate contention model when it comes to modeling flooding-based schemes. A first effort to that direction can be found in [74].

5.3 Adaptive Reinforcement-Based Routing for DTNs

This section introduces a novel encounter-based routing protocol [80] by using Collaborative Reinforcement Learning (CRL)[75, 78] as a self-organizing technique, called Adaptive Reinforcement-Based Routing Protocol (ARBRP). The main feature of the proposed protocol is the strong capability in adaptation to the fluctuation of network status and user behaviors so as reduce the number of transmissions, message delivery time, and increasing delivery ratio. The proposed ARBRP jointly consider node mobility statistics, congestion, and buffer occupancy, which are taken as a feedback in the quality-metric function. In specific, the feedback is in a form of statistical model of estimated contacts reliability based on sampling the availability of channel and buffer space during a contact between nodes. The developed quality-metric targets to facilitating decision making for each active data message, resulting in optimized network throughput.

5.3.1 CLR Model for DTN Routing

In this work, the design of DTN routing protocols is formulated as a reinforcement learning problem, in which the states, actions, transition and reinforcements of the proposed system will be explicitly identified. A learning strategy under the network constraints is then constructed and exercised in the routing protocol.

Background of the Model

In the terminology of CRL, a DTN is modeled as a time-varying environment in which the state at each node is determined by (1) the relative position of nodes, (2) the destination of carried messages, (3) the connectivity information on links between nodes, and (4) the buffer occupancy status. With the considered scenario of the study, the DTN routing protocol design encounters number of challenges, including the lack of global knowledge at any particular node in the DTN, and the requirement for the system-wide autonomic properties of the routing protocol to emerge from local routing decisions at routing nodes.

To trade-off between the system state observability and computation complexity, each node maintains the network status within a fixed time-window, in which the observations on terms of nodal mobility, links activities, and buffer occupancy (or referred to as *contact statistics*) are collected. The contact statistics are further fed as input to the utility function $V(s)$ (also called a cost function or quality-metric in context of DTN), which is specific to every node pairs A and B, and are maintained in a table, called *contact table.* A node A exchanges its contact statistics with a node B when they move into each other's transmission range, and the newly obtained contact statistics will be used to exchange their *contact table.* Meanwhile, the utility function $V_A(s)$ and $V_B(s)$ of a causally connected external state, s, is updated accordingly. These update actions are so-called reinforcements, which continue equipping each node with intelligence and knowledge in decision on message forwarding.

Whether a message should be forwarded from node A to an encountered node B will be determined by the value of utility function $V_B(s)$ maintained at a node B, the history of the quality of wireless link between A and B, and the history of buffer status of node B. A transmission could be failed during a contact due to bad channel condition and contention with other surrounding users, which can consume the time on-the-fly and leaves the messages to remain buffered at the node until a new contact happens. This channel busy-time can cause extra delay in message delivery or increase message loss.

Let the action of handing over a message from node A to B be denoted as $A_{h_{\mathbf{B}}}$. If the message hand-over is successful, the transition $A \longrightarrow B$ occurs, and (or

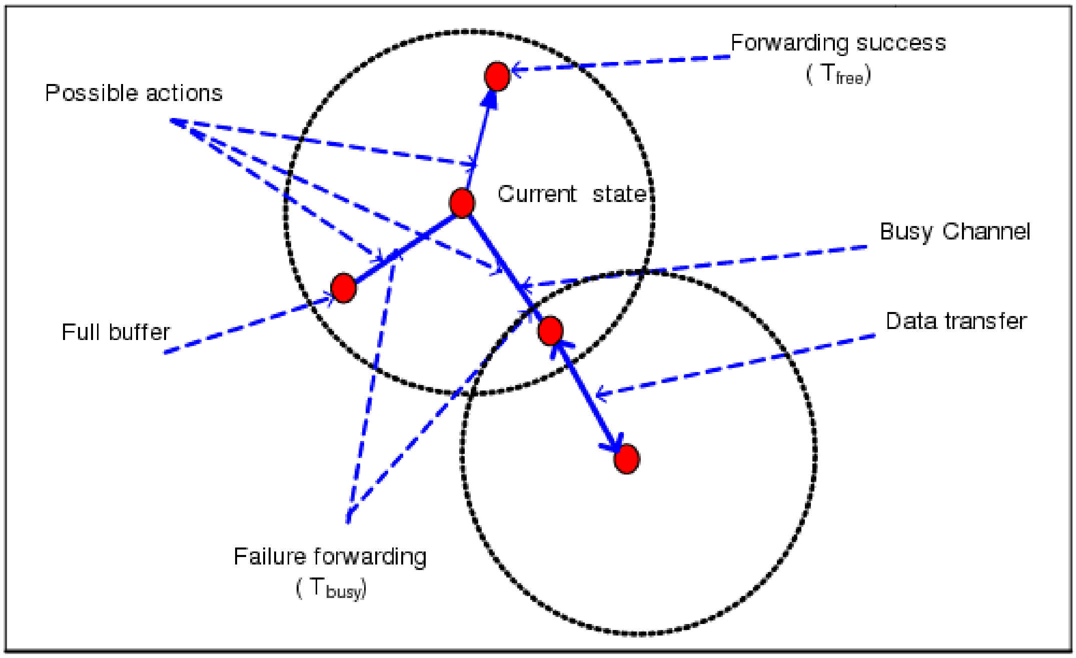


Figure 5.5: The reinforcement model of DTN routing.

$A \longrightarrow A$) otherwise, in which the message remains in state A . Note that a time-to-live (TTL) is defined for each message, and a message is dropped if its TTL is expired.

Model Reinforcements

The cost function should reflect the cost of a given state transition, which depends on the state of the system and the action performed. In the CRL model of DTNs, the choice of forwarding action is mainly based on the routing policy and the utility function used for building information about network behavior.

The reinforcement function (i.e, the utility function) evaluates a given state transition and the corresponding action. The action in the proposed CRL model of DTNs is whether a specific message should be forwarded to the encountered node. Figure 5.5. shows the reinforcements considered for DTN routing. As a number of notations are introduced as follows. Let T_{free} denote the amount of time a channel is available during a time window interval, $W^{(i)}$, and T_{busy} denotes the amount of time that the channel is busy or the buffer of the encountered node is full during a time window interval, $W^{(i)}$. In other words, T_{free} denote the reinforcement of a successful exchange, whereas T_{busy} represents the reinforcement of a failed message exchange attempt. Let $T_{total} = (T_{busy} + T_{free})$ denote the total contact time between any node pairs during $W^{(i)}$. Each node in the network keeps track of the history of these values on every other node in the network, and are calculated at the end of each time window $W^{(i)}$.

5.4 Adaptive Reinforcement Based Routing Protocol (ARBRP)

Based on the reinforcement model, the proposed ARBRP for DTNs is introduced in this section. The components of the functional modules of the ARBRP architecture is similar to that introduced for SAURP in Figure 5.1. The difference is in the way of constructing each model which is introduced in this section.

5.4.1 Protocol Background

In order to introduce the proposed protocol as CRL technique, a number of items; namely, states, delegation actions, transitions, connection cost, and cost function, are introduced as follows.

States definition: Each node A has a set of states $S_A\{M_w, F_s, R_s\}$, where M_w represents a message that is waiting in the buffer for forwarding, F_s represents the event that a messages has been successfully forwarded to an encountered node, and R_s represents the event that the message successfully has been received at node A. Also, A has internal and external states: $Ext(A) = M_w$ and $Int(A) = R_s, F_s$. For $F_s \in Int(S_A)$, there is causally connected state $M_w \in Ext(S_B)$, where node $B \in r_A$, r_A is the set of nodes in the transmission range of node A or nodes that are frequently encountered by node A in a certain time window.

Delegation actions at each node: The set of actions available at node A is $AC = A_{h_A} \cup \{Receive\} \cup \{exchange\ of\ summary\ vectors\}$, where A_{h_A} represents the set of delegation actions, A_{h_A} $=\{H_A(r_0), ...H_A(r_{N-1})$, N is the set of nodes in the transmission range of node A. The action $H_A(B)$ represents an attempt to hand over a message from node A to node B. The action $\{Recieve\}$ represents an attempt to deliver a message to current custodian node A. The $\{exchange\ of\ summary\ vectors\}$ action is used for message vector exchange when two nodes become in transmission range of each other.

The state transition between nodes: The state transitions model for the MDP are as follows:

- The probability of successful message exchange between nodes A and B is $Ps_{AB} = p_A(F_s \mid M_w, a_B)$.
- The probability of failed message exchange is $P_A(M_w \mid M_w, a_B) = 1 - Ps_{AB}$.
- The probability of delivering message to destination if the current node is the destination is $P_A(R_s \mid M_w, deliver) = 1$
- The probability for all other states is $P(s' \mid s, a) = 0$.

5.4.2 Contact Statistics (CS)

The connection cost is affected by T_{free} and T_{busy}. $T_{free} \in V(s)$ represents the cost in case of successful message hand over($M_w \longrightarrow F_s$) under delegation action , and $T_{busy} \in V(s)$ is the cost in case of failed delegation action ($M_w \longrightarrow M_w$), which can be formulated as follows.

- $C_A(M_w \mid F_s, a) = T_{free}$ (time in which channel is free), where $a \in A_{h_A}$.

- $C_A(M_w \mid M_w, a) = T_{busy}$, where $a \in A_{h_A}$. It includes the cases when the buffer of the encountered node is full or the channel is busy.

- $C_A(s' \mid s, a) = 0$, where $a \notin A_{h_A}$

5.4.3 Utility-function Calculation and Update Module (UCUM)

Calculation of Cost (Utility) Function ($V_A(s)^{(i)}$)

Given the estimated models $P(s' \mid s, a)$, $C(s' \mid s, a)$, and $(V_B(s'))$ the optimal value function can be calculated by solving the set of modified Bellman equations [79] for distributed model-based reinforcement learning as follows:

$$V_A(s) = max_a[Q_A(s, a)], \tag{5.21}$$

$$Q_A(s, a) = \sum_{s' \in M_w, F_s, R_s} P_A(s' \mid s, a_A) \left(C_A(s' \mid s, a_A) + (V_B(s'))\right) \tag{5.22}$$

Note that the calculation of Q-value is quite simple since each action has only two possible outcomes. For the event of handing over the message to encountered node B, given the message at buffer of custodian node A (state M_w), the Q-value at the node A is:

$$Q_A(M_w, a_B) = Ps_{AB}\left[T_{free} + V_B(s')\right] - Pf_{AB}\left[V_A(s') + T_{busy}\right]$$

$$= Ps_{AB}\left[T_{free} + V_B(F_s)\right] - Pf_{AB}\left[V_A(M_w) + T_{busy}\right] \tag{5.23}$$

where the minus sign represents the network backward feedback from link to the node, T_{free} represents the duration of time in which the channel is free at node A during time-window period $W^{(i)}$, T_{busy} is the duration of time in which the channel is busy during time-window period $W^{(i)}$. $Ps_{AB} = \frac{T_{free}}{T_{total}}$ is the probability that the link between node A and node B is available during time-window$W^{(i)}$. $Pf_{AB} = 1 - Ps_{AB}$ is the probability that the link is busy during time-window $W^{(i)}$. The buffer effect takes place when the buffer of the encountered node is full. In this case, the time in which the buffer is full during a contact is added to the busy time of the channel, i.e., the busy time composed of the channel busy time and the time in which the buffer of the encountered node is full. The busy-time must be represented as "backward feedback" in the cost function Q. Note that the Q value is calculated at each custodian node regarding to the destination node, then the custodian node will make the decision of forwarding according to the routing forwarding policy. This routing strategy is mainly based on $max_a Q(M_w, a_B)$ values which can be simplified as:

$$Q_A(M_w, a_B) = \frac{Ps_{AB}(T_{free} + V_B(F_s)) - Pf_{AB}.T_{busy}}{1 - Pf_{AB}}$$

$$= V_B(F_s) + T_{free} - \frac{Pf_{AB}}{Ps_{AB}} T_{busy} \tag{5.24}$$

where $V_B(F_s)$ is quality-metric, (cost function) represents the average cost of contact time between node B and the destination node D. Note that $V_A(M_w)$ is set to zero since we need to consider only the advertised $V_B(F_s)$ from node B regarding the destination node.

Time-window update

As discussed earlier, the connectivity between any two nodes is measured as the amount of meeting time intervals during a time-window $W^{(i)}$, which is mainly based on the time in which the wireless channel is busy or the buffer is full, and the time in which the channel is free and the buffer is available. These contact period components are time varying. They can change largely from time window to time window which have huge impact on the protocol forwarding policy. Hence, a representative smooth transfer of $V(s)$ values between consecutive time-windows is needed. We propose adaptive parameter that reflects the rate of change of connectivity between nodes and can be used to make smooth transfer from a previous time window to a current time window. We use T_{total1} and T_{total2} to refer to the total contact time (T_{total}) durations obtained during time windows $W^{(i-1)}$ and $W^{(i)}$, respectively. Three cases are identified: 1) the contact time duration in the previous time-window is less than the current time-window; 2) the contact time duration in the previous time-window is greater than current time-window; 3) no contact happened during one of the time-windows. The accumulated total contact time at the end of current time window is based on both the total free contact time durations in current time-window, $V_{cs}(s)^{(i)}$, and the last accumulated total contact time obtained in previous time-window, $V(s)^{(i)}$. To combine these changes of connectivity values when transferring from previous time window to current time window, the three cases are formulated as follows:

- if $T_{total1} > T_{total2}$

$$V(s)^{(i+1)} = (1-\gamma)V(s)^{(i)} + \gamma.V_{cs}(s)^{(i)} \tag{5.25}$$

- if $T_{total1} < T_{total2}$

$$V(s)^{(i+1)} = \gamma.V(s)^{(i)} + (1-\gamma)V_{cs}(s)^{(i)} \tag{5.26}$$

  The parameter γ is given by

$$\gamma = \frac{\mid T_{total1} - T_{total2} \mid}{max(T_{total1}, T_{total2})}, T_{total1}, T_{total2} > 0 \tag{5.27}$$

- if $T_{total1} = 0$, or $T_{total2} = 0$, then $V(s)^{(i+1)} = \frac{V(s)^{i-1} + V_{cs}(s)^{i}}{4}$. This case represents worst case scenario, i.e. unstable node behavior, or low quality of node mobility. Hence, the $V(s)^{i+1}$ value should be low. $V(s)^{(i+1)}$ represents the routing metric value that is used as input to the next time window. This value is maintained as a vector of contact time that is specific to every other node, and the vector is called *routing metric table.* The routing metric table can be employed in the decision making process for message forwarding.

5.4.4 Transitivity Update Module (TUM)

When two nodes are within their transmission range, they exchange V vectors regarding the message destination. With the update, the custodian node decides whether or not the message should be forwarded to the encountered node. This exchange of summary vectors is followed by transitivity update which has the similar idea of that introduced in section (5.2.3). Thus, the congestion history of buffer and link availability of node B plays a key role in using transitivity property. In order to maximize the average contact time, $V(s)$, between node A and D and make any message destined to node D goes through node B, a proper update using transitivity property should be made. To deal with this maximization problem (maximizing $V(s)$) using transitivity update, we formulate the information presentation first as inter-contact time [63], then the corresponding average contact time is obtained. Using inter-contact time in the transitivity update is simpler and can adaptively update values only when $V_{(A,D)} < V_{(B,D)}$ in order to ensure that node A reaches D through B. Otherwise, if $V_{(A,D)} > V_{(B,D)}$, the transitivity property is not useful since node A is a better candidate for forwarding messages directly to node D rather than forwarding them through B. The inter-contact time between nodes A and D, $\triangle T_{(A,D)}$, is calculated by $\frac{W^{(i)}}{V^{(i)}_{(A,D)}}$, where $W^{(i)}$ represents duration of sliding window, $V^{(i)}_{(A,D)}$ is the average contact time duration ($V_A(s)$) between A and D during $W^{(i)}$. $\triangle T^{(i)}_{(A,B)}$, and $\triangle T^{(i)}_{(B,D)}$, are obtained using the similar way. The new updated inter-encounter time is calculated as follows:

$$\triangle T^{(new)}_{(A,D)} = \alpha \triangle T^{(i)}_{(A,D)} + (1-\alpha)(\triangle T^{(i)}_{(A,B)} + \triangle T^{(i)}_{(B,D)}) \tag{5.28}$$

where α is a weighting factor calculated from

$$\alpha = \frac{\triangle T^{(i)}_{(A,B)} + \triangle T^{(i)}_{(B,D)}}{\triangle T^{(i)}_{(A,D)}} \tag{5.29}$$

Note that α must be less than 1; that is $\triangle T^{(i)}_{(A,D)} > \triangle T^{(i)}_{(A,B)} + \triangle T^{(i)}_{(B,D)}$.

The new contact time is obtained by applying the following relation:

$$V^{(new)}_{(A,D)} = \frac{W^{(i)}}{\triangle T^{(i+1)}_{(A,D)}} \tag{5.30}$$

$V^{(new)}$ represents the new values of $V(s)$ that is obtained form the transitivity update. The introduced transitivity-update rule has great impact on protocol performance.

5.4.5 The Forwarding Strategy Module (FSM)

The Weighted Copy Rule

The forwarding of message copies is based on the goodness of the encountered node regarding the destination [63]. The source of a message initially starts with L copies; any node A that has N_A >1 message copy tokens (source or relay) and that encounters another node B with no copies, hands over to node B a number of message copy tokens according to its goodness for the destination node D. Node A hands over some of the message copy tokens to node B and keeps the rest for itself according to the following formula:

$$N_B = \left\lceil \frac{V_{(A,D)}.N_A}{V_{(A,D)} + V_{(B,D)}} \right\rceil \tag{5.31}$$

where N_A is the number of message tokens that node A has, $V_{(B,D)}$ and $V_{(A,D)}$

are the average of contact time between node B and node D, and node A and D, respectively. Node A hands over message tokens to node B only if the value of $V(s)$ at node B regarding to the destination is better than that of node A. This formula guarantees that the largest number of message copies is spread to relay nodes that have better information about destination node. After L messages have been copied to custodian nodes, each of the L nodes carrying a copy of the message performs according to the forwarding rule as descried in the following section.

The Forwarding Rule in ARBRP

- If the destination node is one hop away from an encountered node, the custodian node hands over the message to the encountered node.
- If more than one node exist in the transmission range of custodian node A, a node with the highest value of Q will be chosen among all other nodes, according to the relation $V_A(s) = max_a[Q_A(s, a)]$.
- If the value of $V(s)$ of the encountered node regarding to the destination node is greater than that of the custodian node by threshold value, $Th1$, a custodian node hands over the message to the encountered node.

5.5 Performance Evaluation

To evaluate the SAURP and ARBRP, a DTN simulator similar to that in [51] is implemented. The simulations are based on two mobility scenarios; a synthetic one based on community based mobility model (CBMM) [50], and a real-world encounter traces collected as part of the Infocom 2006 experiment, described in [104].

The performance of SAURP and ARBRP is examined under different network

Algorithm 5.2 The forwarding strategy of ARBRP

On contact between node A and B

Exchange summary vectors

for every message M at buffer of custodian node A *do*

    if destination node D in transmission range of B *then*

        A forwards message copy to B

    end if

    else if $V^{(i)}_{(B,D)} > V^{(i)}_{(A,D)}$ *do*

        if message tokens >1 *then*

            apply weighted copy rule

        end if

        else if $V^{(i)}_{(B,D)} > V^{(i)}_{(A,D)} + T_{th}$ *then*

            A forwards message to B

        end else if

    end else if

end for

node connectivity and is compared to SARP and some previously reported schemes listed below.

- Spray and Focus (S&F) [52]
- Epidemic routing [29]
- Delegation forwarding (DF) [62]
- Self- Adaptive routing protocol (SARP) [63]
- Self-Adaptive utility-based routing protocol (SAURP) [85]
- Adaptive reinforcement based routing protocol (ARBRP) [64]

For all the protocols, an attempt has been made to tune the parameters in each scenario separately, in order to achieve good transmission-delay trade offs.

5.5.1 CBMM Scenario

Evaluation Scenarios

In the simulation, 110 nodes move according to the community-based mobility model [50] in a 600 x 600 2-dimensional torus in a given geographical region. The message inter-arrival time is uniformly distributed in such a way that the traffic can be varied from low (10 messages per node in 40,000 time units) to high (70 messages per node in 40,000 time units). The message time to live (TTL) is set to 9,000 time units. Each source node selects a random destination node, begins generating messages to it during simulation time.

The nodes are classified into four groups: (1) 25% of the nodes move locally most of the time (pl [0.8, 0.9]) but may occasionally roam into other preferred communities (pr [0.05, 0.15]); (2) 25% of the nodes move only locally inside their own community(pl = 1, pr = 0); (3) 35% of the nodes quite often roam outside their community (pl = [0.2, 0.3] and pr = [0.5, 0.7]); (4) 15% of the nodes are static and uniformly distributed in the network. Note that this scenario is different to that employed in chapter 4.

The performance of the protocols is evaluated with respect to the low transmission range and varying buffer capacity under high traffic load. Second, with respect to the moderate-level of connectivity and varying traffic load. Third, the performance of the protocols is examined in terms of the bandwidth. Finally, the performance of the protocols is examined in terms of the level of connectivity changes.

The Effect of Buffer Size

In this scenario the performance of SAURP and ARBRP regarding different buffer sizes is examined under a low transmission range (i.e., K = 30) and a high traffic

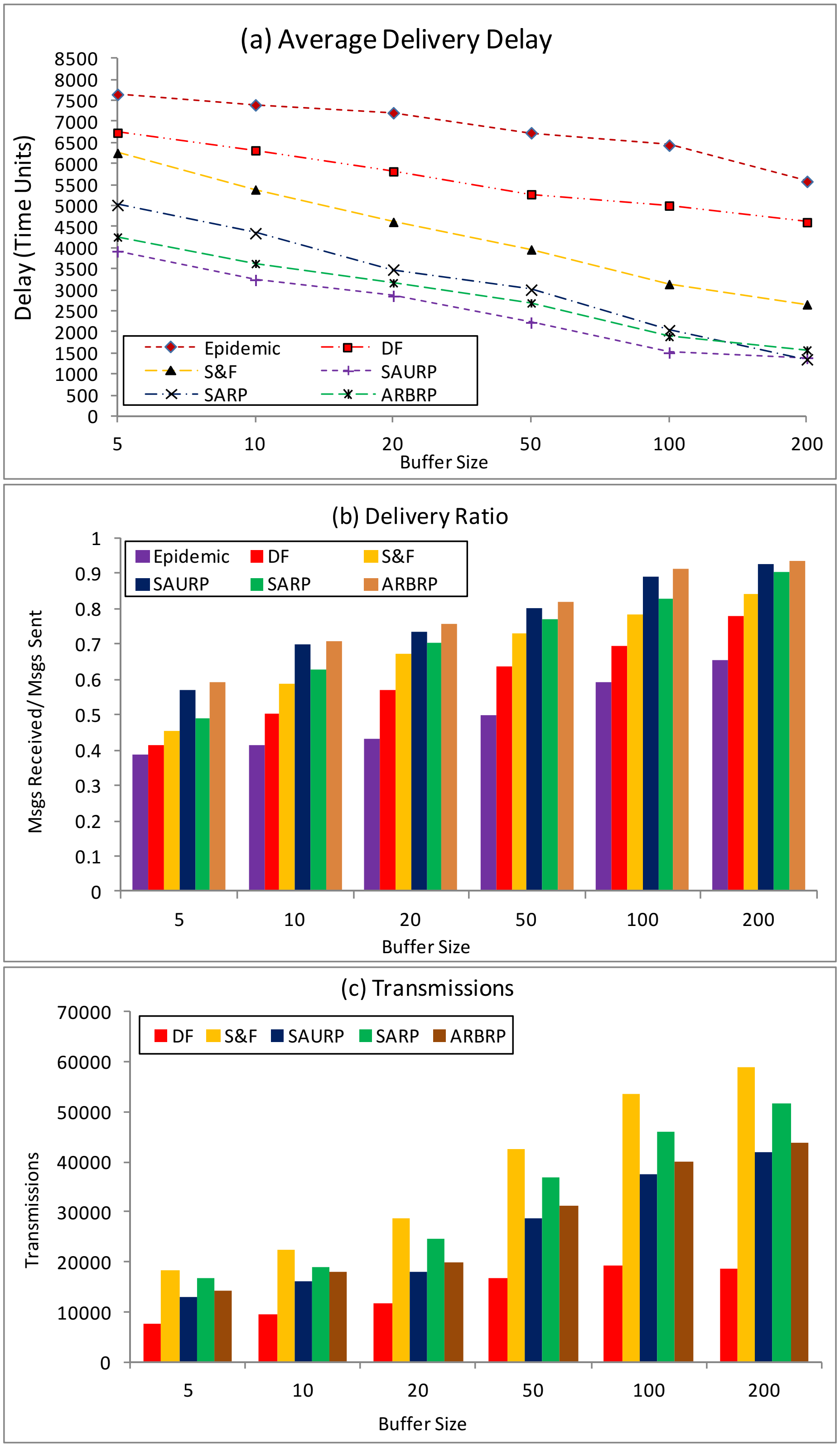


Figure 5.6: The effect of buffer size

load (i.e., 70 messages generated per node). Due to the high traffic volumes, we expect to see a significant impact upon the message forwarding decisions due to the degradation of utility function values caused by buffer overflow. Note that when the buffer of the encountered node is full, some messages cannot be delivered even though the encountered node metric is better than the custodian node. This situation results in extra queuing delay, especially in the case that flooding-based schemes are in place. Figure 5.6 shows the experiment results where the buffer space was varied from 5 (very limited capacity) to 200 (relatively high capacity) messages to reflect the performance of the protocols under the considered traffic load. As shown in Figure 5.6, when the buffer size is small (50 messages or less) the performance of the protocols is very sensitive to the change of buffer capacity.

It is observed that Epidemic routing produced the worst delivery delay in all scenarios, since it has been critically affected by both the limited buffer size and mobility model. On the other hand, since SAURP and ARBRP take the situation that a node may have a full buffer into consideration by degrading the corresponding utility metric, they produced the best performance. In specific, both SAURP and ARBRP yielded the shortest delivery delay with superior performance by SAURP. For example, when the buffer size is 50 messages, SAURP achieved shorter delivery delay than ARBRP by 19%, SARP by 32%, DF by 230%, and S&F by 73%. Although SUARP and ARBRP produced more transmissions than DF, they still less than that produced by S&F, and SARP.

Regarding the delivery ratio, ARBRP and SAURP can achieve the best performance compared to all other schemes with superior performance by ARBRP, while the epidemic routing degrades below 50% for high traffic loads. ARBRP can achieve delivery ratio above 83%, SAURP 80%, SARP 76, S&F 73, and DF 63%. As the buffer size increased, the performance of all protocols was improved especially for SARP. When the buffer size is larger than the traffic demand, the SARP scheme yielded the best delivery delay among all other schemes due to the relaxation of buffer capacity limitation. SAURP and ARBRP still yielded competitive delay and best delivery ratio with a smaller number of transmissions than S&F.

The Effect of Traffic Load

The main goal of this scenario is to observe the performance impact and how SAURP and ARBRP react under different degrees of wireless channel contention. The network connectivity is kept high (i.e., the transmission range is set to as high as 70 meters) under different traffic loads, while channel bandwidth is set relatively quite small (i.e., one message transfer per unit of time) in order to create an environment with non-trivial congestion. We have two scenarios for nodal buffer capacity: 1) unlimited capacity; and 2) low capacity (10 messages). Figure 5.7 shows the performance of all the routing algorithms in terms of the average delivery delay, delivery ratio, and total number of transmissions.

It is observed that Epidemic routing produced the largest delivery delay and requires a higher number of transmissions compared to all the other schemes, thus it is not included in the figure. Note that the Epidemic routing is subject to at least 3 times of longer delivery delay than that by S&F and an order of magnitude more transmissions than that by SUARP.

As shown in Figure 5.7, when the traffic load is increased, the available bandwidth is decreased accordingly, which causes performance reduction. When the traffic load is moderate (i.e., less that 50 messages per node), it is clear that the delivery delay is short in all the schemes. SAURP outperforms all other protocols and ARBRP is the second best. SAURP can produce delay shorter than that of ARBRP, SARP, DF, and S&F by 26%, 55%, 323%, and 250%, respectively. Regarding the delivery ratio, SAURP, ARBRP, SARP, and S&F can achieve excellent performance of 95%, while the epidemic routing degrades below 40% for high traffic

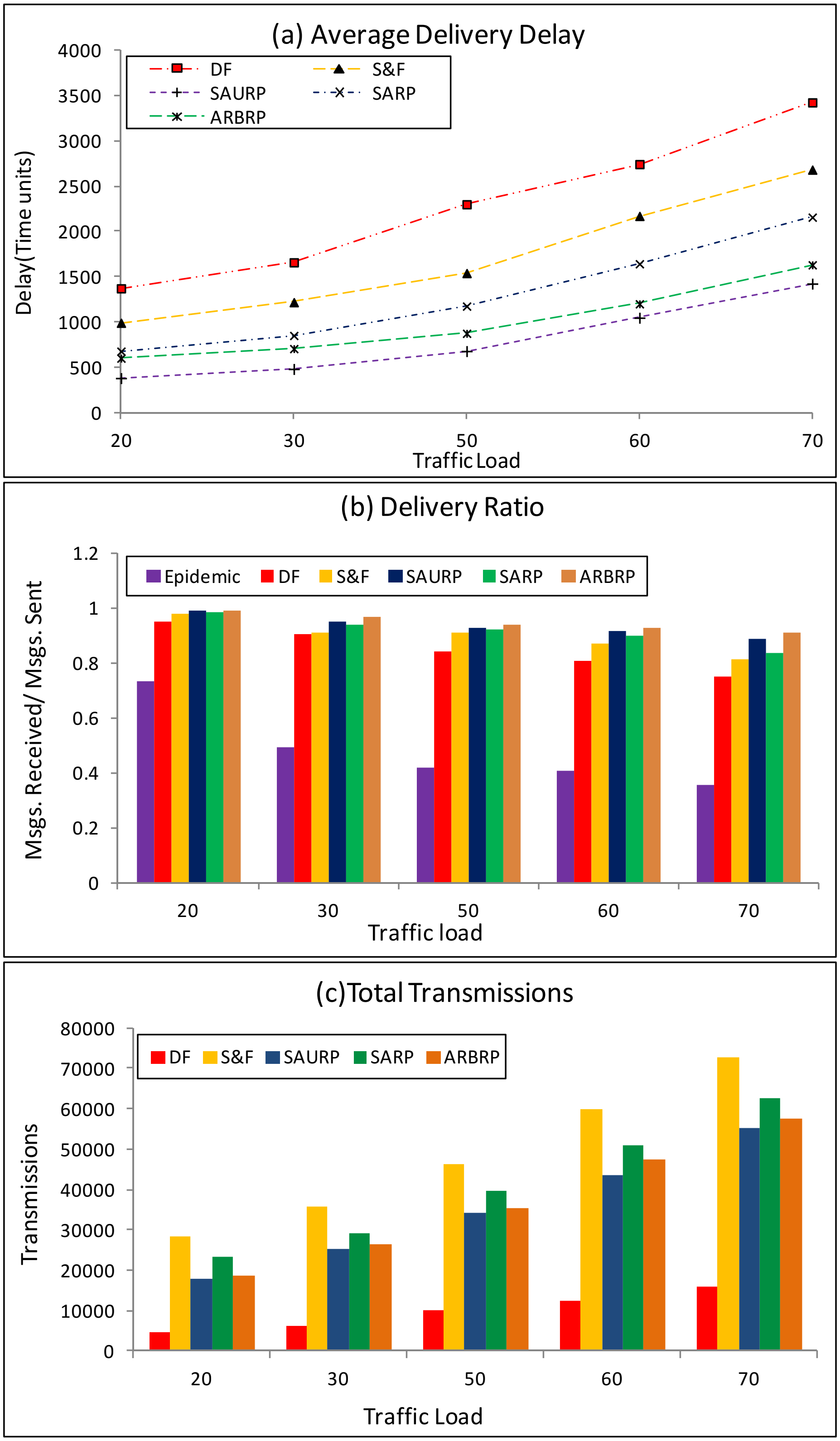


Figure 5.7: The effect of traffic load under high buffer capacity

loads. DF can achieve delivery ratio above 91%.

As expected, the performance of all the schemes degrades as the wireless channel contention is getting higher, especially when the traffic load exceeds 50 messages per node. We observed that SAURP and ARBRP can achieve significantly better performance compared to all the other schemes, due to the consideration of busy links in their message forwarding mechanism, where the corresponding routing-metric is reduced accordingly. This results in the ability of rerouting the contended messages through the areas of low congestion. In summary, the delivery delay obtained by the SAURP in this scenario is shorter than that of the ARBRP by 16%, SARP by 53%, S&F by 210%, and DF by 257%, respectively. Regarding delivery ratio, ARBRP can achieve as high as 91%, compared to 88% by SAURP (the second best), 82% by S&F, and 76% by DF. Even though DF produced the lowest number of transmissions, it is at the expense of the worst delivery delay and delivery ratio.

As the buffer capacity is low (e.g., 15 messages) and the traffic load is high, the available bandwidth decreases and the buffer occupancy increases accordingly, which makes the performance of all protocols degraded. It is notable that SAURP and ARBRP outperform all the their counterparts in terms of delivery delay and delivery ratio under all possible traffic loads. When the traffic load is high, SAURP yielded shorter delivery delay than that of ARBRP by 17%, SARP by 63%, S&F by 90%, and DF by 260%. Although SAURP and ARBRP require more transmissions compared to DF, the number is still smaller than that produced by S&F. For high traffic loads, ARBRP can achieve delivery ratio up to 90%, and SAURP up to 88%, while the epidemic routing, SARP, DF, and S&F degrade to 38%, 70%, 53%, and 66%, respectively. Figure 5.8 shows the performance of all techniques under this scenario.

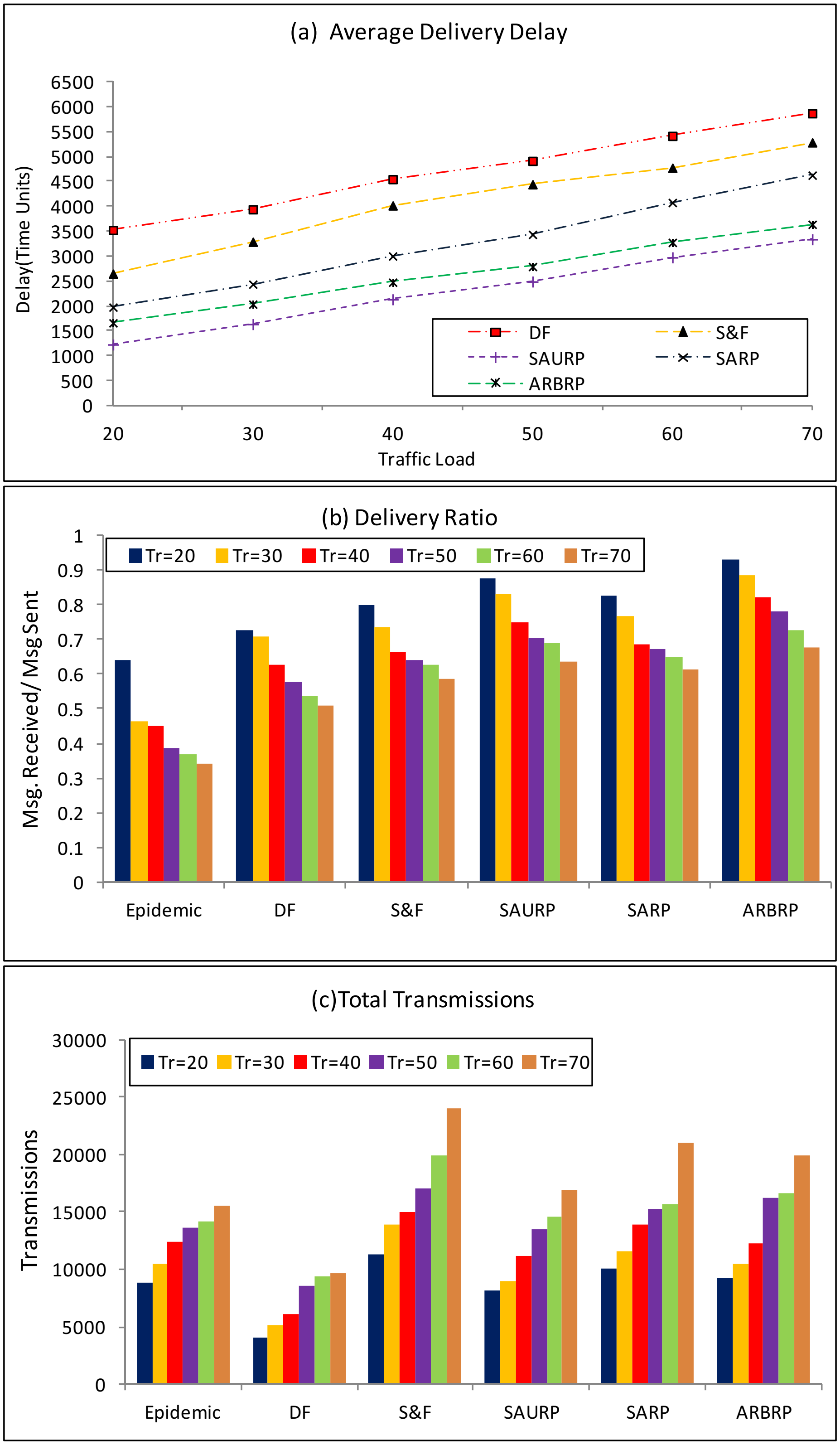


Figure 5.8: The effect of traffic load under low buffer capacity

The Effect of Channel Bandwidth and Traffic Load

To examine the effect of channel bandwidth, the network connectivity is set to moderate (under moderate transmission range by setting K = 50), and the link capacity is set five times higher than that used in the previous scenarios in order to avoid bottlenecks in the traffic loads. Figure 5.9 shows the performance of all the routing protocols in terms of the average delivery delay, delivery ratio, and total number of transmissions.

As the link bandwidth increased, it can be seen from Figure 5.9 that the epidemic routing achieves the best performance with respect to delivery delay, because the buffer capacity is unlimited and the contention on the bandwidth is relaxed. SARP achieves the second best performance compared to the other schemes. It outperforms SAURP and ARBRP schemes. One explanation is that SARP mechanism considers the uncertainty that may result when the utility function values between encountered nodes are close to each other (i.e. the difference between the value of the utility function at the custodian node and the encountered node regarding the message destination is less than the threshold value required to perform the forwarding decision). SAURP and ARBRP still have very competitive delivery delay. SAURP has a shorter delay than ARBRP by 13 %, S&F by 56%, and DF by 266%. It has longer delay than epidemic by 30%, and SARP by 16%. Meanwhile, SAURP and ARBRP needed less transmissions compared to that by S&F and SARP. Even though DF produced the lowest number of transmissions, it has the worst performance in terms of delivery delay and delivery ratio. All protocols achieved a delivery ratio above 90%. SAURP and ARBRP maintain high delivery ratio: above 96%.

The above results show that channel bandwidth has significant impact on the performance of the protocols. If the available bandwidth is much higher than the total traffic load, flooding based schemes [11] can yield delivery delay as SAURP

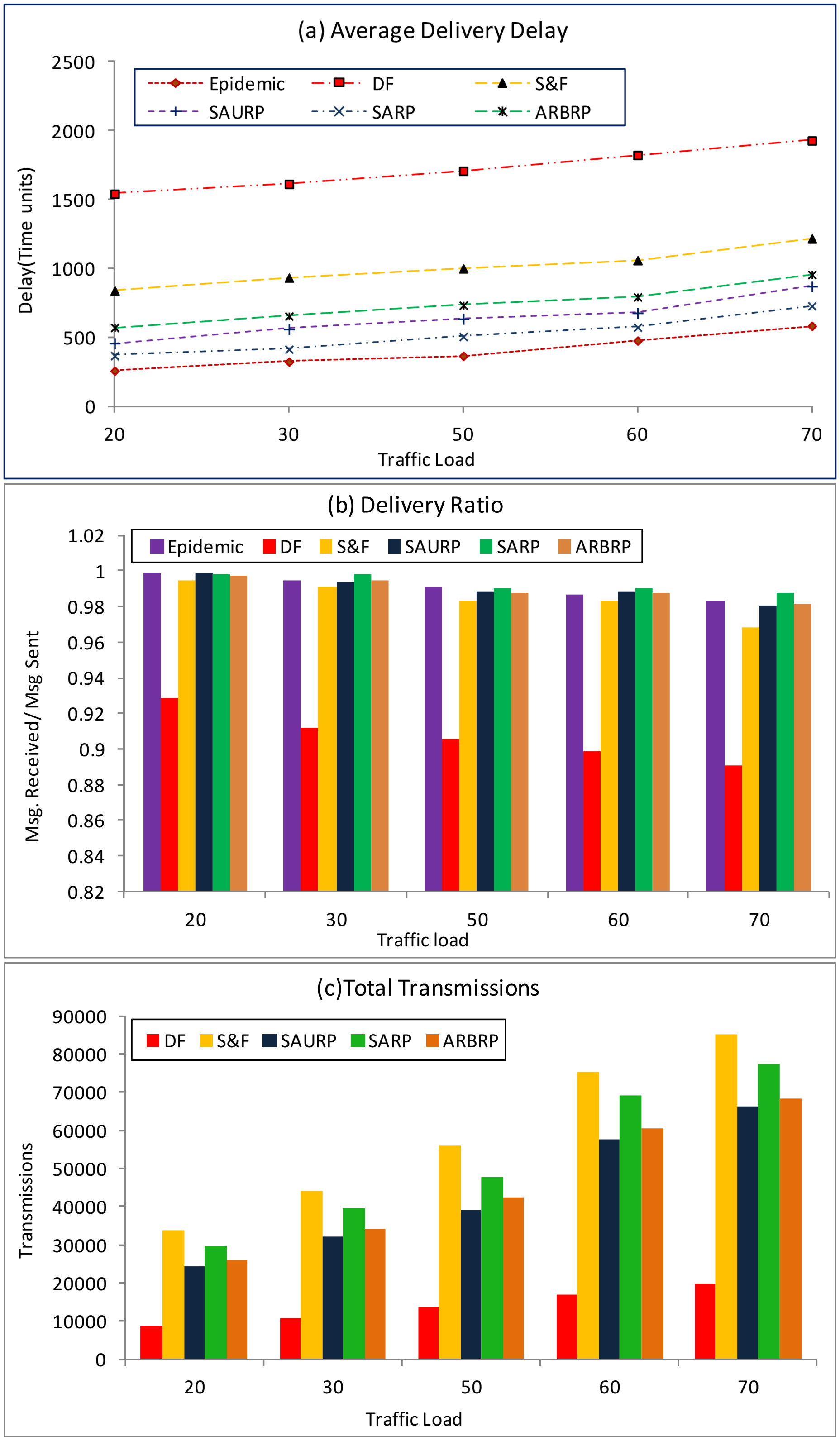


Figure 5.9: The effect of traffic load under high link bandwidth

at the expense of taking far more transmissions. On the other hand, if the channel bandwidth is limited, SAURP, ARBRP, and the spraying schemes outperform the flooding-based schemes because of the contention caused by limited bandwidth.

The Effect of Connectivity

This scenario studies the performance impact due to network topology connectivity. In the scenario, the level of connectivity is increased from very sparse to highly connected by varying the value of K while observing the resultant impact on the performance. We are particularly interested to investigate the SAURP and ARBRP mechanism in response to heavy traffic loads which result in high contention on the wireless channel. The buffer capacity is kept low (15 messages), and the traffic load is considerably high (70 messages). Figure 5.10 shows the average delay, delivery ratio, and the number of transmissions as a function of transmission range.

It is observed that although epidemic routing takes the most transmissions, it is still far outperformed by other schemes in terms of delivery delay, thus it is not included in the plots. SAURP and ARBRP outperform all the schemes in terms of delivery delay while taking noticeably fewer transmissions than that by $S\&F$ and SARP under low-level of connectivity. When the network is sparsely connected, SAURP and ARBRP can achieve shorter delivery delay than all other schemes, that is because the performance of other schemes is affected by the uncertainty of buffer occupancy status. On the other hand, when the network is moderate-connected, SARP can achieve a competitive-level of delivery delay to SAURP and ARBRP with more transmissions. As the network becomes almost connected and the traffic load is high, the uncertainty of both buffer occupancy status and the availability of bandwidth affect the performance of the other techniques. As a result, SAURP and ARBRP outperform all other schemes in terms of delivery delay and delivery ratio with superior performance to ARBRP in terms of delivery ratio.

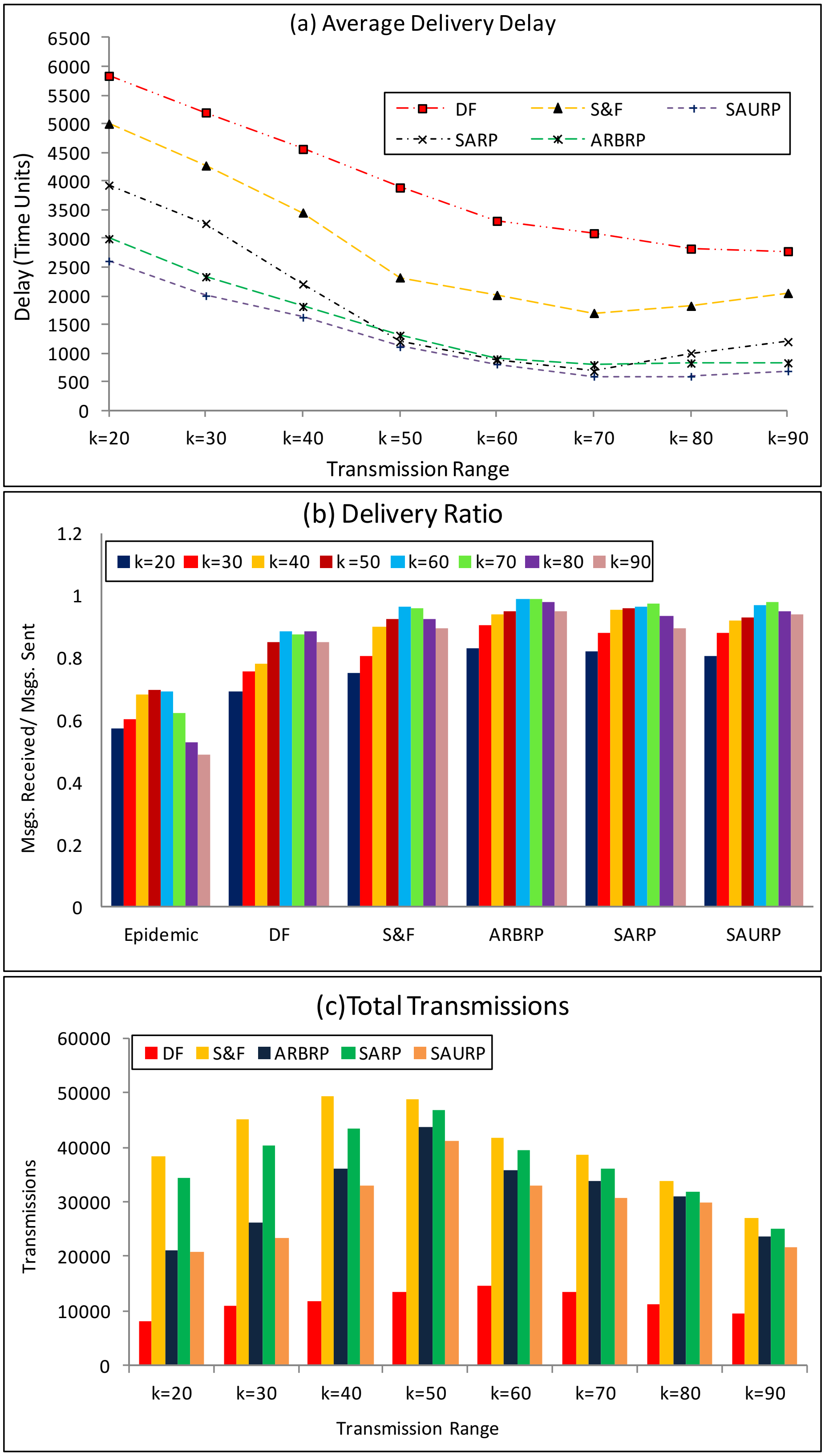


Figure 5.10: The effect of connectivity

5.5.2 Real Trace Scenario

In order to evaluate SAURP and ARBRP in realistic environment, the performance of the schemes is examined using real encounter traces. These data sets comprise of contact traces between short-range Bluetooth enabled devices carried by individuals in Infocom 2006 conference environment. More details about the devices and the data sets, including synchronization issues can be found in [104]. In order to observe the performance impact and how SAURP reacts under congested environment, we set the bandwidth, buffer capacity, and the distribution of the contact time such that congested environment is formed. The channel bandwidth is set relatively quite small (i.e., one message transfer per unit of time), and the buffer size is set to 10, under different levels of traffic demand.

Figure 5.11 shows the performance of all the routing algorithms in terms of the average delivery delay, delivery ratio, and total number of transmissions.

As the buffer capacity is low and the traffic load is high, the available bandwidth decreases and the buffer occupancy increases accordingly, which degrades the performance of all protocols, especially for the Epidemic routing and. It is observed that Epidemic produced the largest delivery delay. It is subject to at least 2.7 times of longer delivery delay than that by SARP. It is notable that SAURP outperforms all the multiple-copy routing protocols in terms of delivery delay under all possible traffic loads, while ARBRP outperforms all other schemes in terms of delivery ratio. When the traffic load is high, SAURP yielded shorter delivery delay than that of ARBRP by 10%, SARP by 22, SF by 30%, and DF by 40%. Although SAURP and ARBRP require more transmissions compared to DF, the number is still smaller than that produced by S&F. For high traffic loads, ARBRP can achieve delivery ratio up to 70%, and SAURP up to 67%, while the epidemic routing, SARP, DF, and S&F degrade to 38%, 64%, 51%, and 59%, respectively.

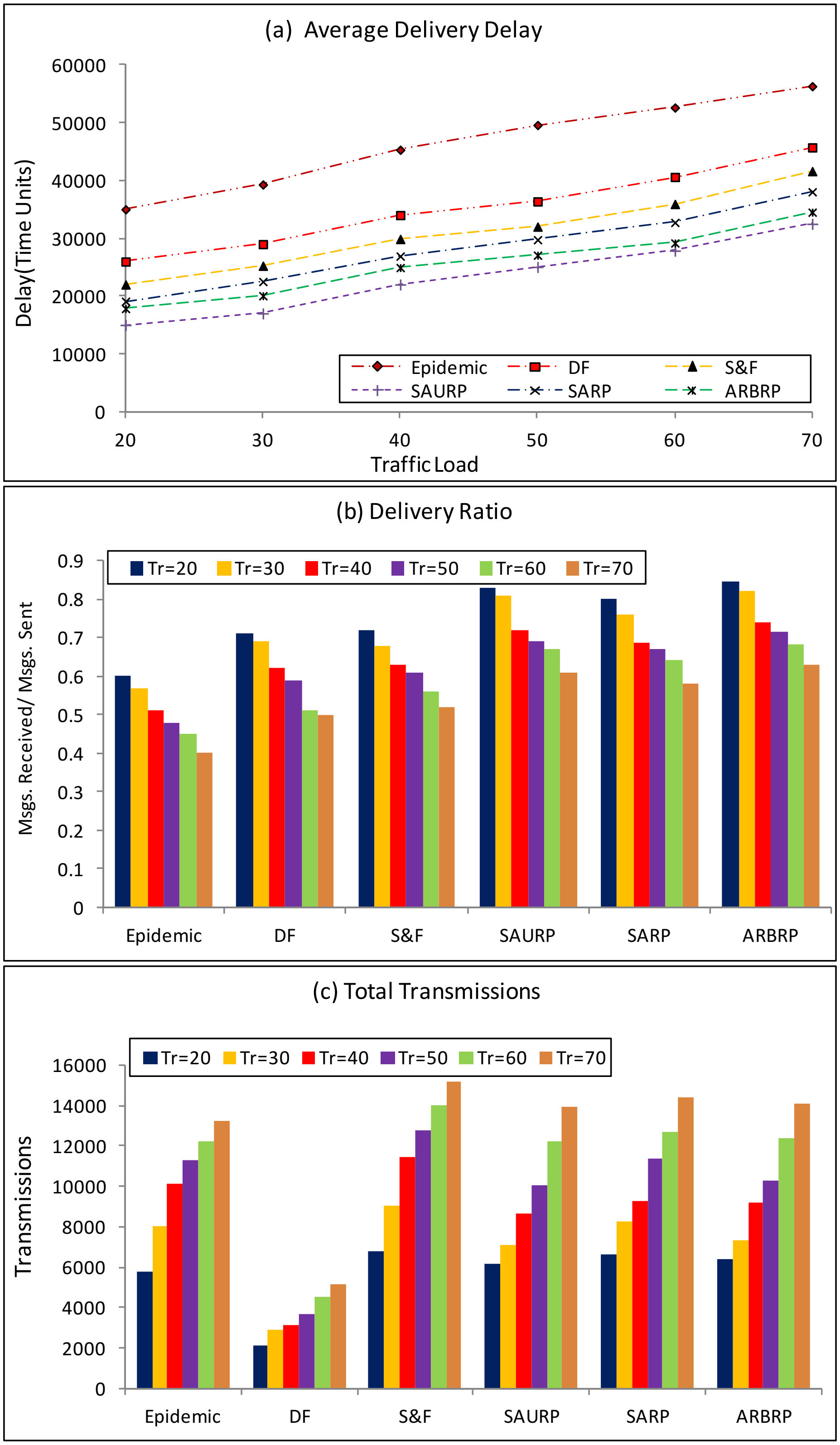


Figure 5.11: The effect of traffic load under trace based scenario

5.6 Summary

This Chapter introduced two novel multi-copy routing schemes called SAURP and ARBRP, for intermittently connected mobile networks. SAURP and ARBRP are characterized by the ability of identifying potential opportunities for forwarding messages to their destinations via a novel utility function based mechanism, in which a suite of environment parameters, such as wireless channel condition, nodal buffer occupancy, and encounter statistics are jointly considered. Thus, SAURP and ARBRP can reroute messages around nodes experiencing either high buffer occupancy, wireless interference, or congestion, while taking considerably smaller number of transmissions. Each scheme utilized the environment parameters of the network differently. SAURP employed the inter-contact time as the main in put to its utility function which resulted in better performance improvements regarding to the delivery delay, while ARBRP employed the average contact time as the main input to its utility function which resulted in better performance improvements regarding to the delivery ratio. We verified the proposed schemes via extensive simulations and compared them with a number of counterparts. Both schemes have shown great stability and achieved shorter delivery delays and delivery ratio than all the existing spraying and flooding based schemes when the network experiences considerable contention on wireless links and/or buffer space.

Chapter 6

Buffer Management and Scheduling in DTNs

In this chapter we introduce and evaluate a novel buffer management and scheduling framework for routing schemes belonging to two different families; i) Flooding based routing, which are developed to suit the homogeneous nodal mobility, where all nodes are independent and identical distributed (iid). Two widely employed DTN routing schemes are considered in this study, namely; the epidemic (flooding), and controlled flooding (source forwarding) schemes [29, 5], ii) Encounter-based (utility-based) routing, which are designed for heterogeneous nodal mobility. SAURP is considered in this study since it is proven to be effective when the nodal mobility is heterogeneous. An important issue in such category of DTN routing is when and to which node the stored messages should be forwarded. Obviously, both the above routing families require additional features in order to incorporate with the given buffer space and contact capacity limitations at each node.

6.1 Buffer Management and Scheduling for Flooding-based Routing

This section introduces a novel buffer management and scheduling framework for epidemic and two-hop (source) forwarding routing, aiming to enable an effective decision process on which messages should be forwarded and which should be dropped when the buffer is full. Such a decision is made by evaluating the impact of dropping each buffered message according to collected network information.

To cope with the high computation complexity in directly solving a Markov chain model [81, 33], we develop a fluid flow limit model and the corresponding ODE formulation as our solution. The use of ODEs, although serving as an approximation of the Markov chain result, can nonetheless improve the computation efficiency and provide a closed-form expression. Further, the formulation with the proposed fluid flow limit model is highly scalable to the network size, where the complexity does not increase with the number of network nodes. For example, the problem in [86] that takes up to 178 seconds by solving a continuous-time Markov chain can be solved by an equivalent ODE model with only 2.8 seconds; and it shows a dramatically increase of computation complexity by using Markov chain when the problem state space is getting larger, while the number of corresponding ODEs is constant regardless of the number of components in the system.

The ODE solution gives per-message utility values, which are calculated based on the estimation of two global parameters: the number of message copies, and the number of nodes which have "seen" this message (the nodes that have either carried the message or rejected the acceptance of this message). The per-message utility values are calculated at each node and then used for the decision on whether the buffered messages should be dropped in any contact. We will demonstrate a closed-form solution to the proposed ODE approach, such that each per-message utility can be calculated efficiently. Simulation is conducted and the results confirm the efficiency and effectiveness of the proposed buffer management scheme under the epidemic and two-hop forwarding routing.

6.1.1 Background and System Description

This subsection presents the background of our mathematical model as well as the network model for encounter-based epidemic routing.

6.1.2 Background of Fluid Flow Model

In a nutshell, this part of the thesis formulates the buffer management task in DTN epidemic and source forwarding routing as a fluid-flow Markov-chain process, respectively. The fluid flow model can then be used to formulate the rate of message propagation among nodes, calculating the expected time until a given node (destination) is infected, and then calculating the delivery ratio (delivery probability). Since solving the fluid flow model using a Markov chain based approach is subject to extremely high computation complexity, we approximate the problem by using an ordinary differential equation (ODE) and derive a close-form solution of the problem. Note that the ODE based approach for solving a Markov chain model has been used for similar problems in the literature [54, 33] with proved efficiency and correctness. The following notation are used throughout this part of this chapter.

- $n_i(t)$ denotes the number of nodes with message i in their buffers (also referred to as "infected" at time t), where t is counted from the creation time of message i. The following relation is used to calculate$n_i(t)$:

$$\frac{dn_i(t)}{dt} = \beta n_i(t)(N - n_i(t)) \tag{6.1}$$

  where N is the number of nodes in the network, and β is the meeting time rate between nodes. Solving (6.1) with the initial condition $n_i(0)$ yields

$$n_i(t) = \frac{Nn_i(0)}{n_i(0) + (N - n_i(0))e^{-\beta Nt}} \tag{6.2}$$

- $P_i(t) = P_i(T_d < t)$ denotes the cumulative probability (CDF) of message i being delivered at time t, where T_d denotes a random variable for the time instant that the message i is successfully delivered. $P_i(t)$ can be expressed in a differential equation form [33]:

$$\frac{dP_i(t)}{dt} = \beta n_i(t)(1 - P_i(t)) \tag{6.3}$$

solving (6.3) with the initial condition $P_i(0) = 0$ yields

$$P_i(T_d < t) = 1 - \frac{N}{N - n_i(0) + n_i(0).e^{\beta Nt}} \tag{6.4}$$

(6.2) and (6.4) are valid only for unlimited buffer space. To extend the above relations to the scenario with limited buffer space, an additional factor should be considered (denoted as P_{fi}), which represents the probability that the encountered node's buffer space is available and the message can be transferred. Note that P_{fi} can be obtained by historical data of nodal encounters. Accordingly, (6.1) is formulated as

$$\frac{dn_i(t)}{dt} = P_{f_i}\beta n_i(t)(N - n_i(t))) \tag{6.5}$$

Thus (6.2) and (6.4) are reformulated as follows:

$$n_i(t) = \frac{N}{n_i(0) + (N - n_i(0))e^{-P_{f_i}\beta Nt}} \tag{6.6}$$

$$P_i(T_d < t) = 1 - \left(\frac{N}{N - n_i(0) + n_i(0)e^{\beta P_{fi} Nt}}\right)^{\frac{n_i(0)}{P_{fi}}} \tag{6.7}$$

Table 6.1: Notation

Variables	Description
$Sr(t)$	The source of message i
$Dst(t)$	The destination of message i
T_i	Elapsed time since the creation of the message
Tx	Time-to-live of message i
R_i	Remaining lifetime of the message ($R_i = Tx_i - T_i$)
$n_i(t)$	Number of copies of message i
$m_i(t)$	Number of nodes who have "seen" message i
$s_i(t)$	Number of nodes who have seen message i and their buffers were not full
P_{f_i}	Probability of forwarding message i to every encountered node

6.1.3 Network Model

In this work, a homogeneous DTN is modeled as a set of N nodes, all moving according to a specific mobility model in a finite area, where inter-encounter time between each pair of nodes is an independent and identically distributed (iid). Let the number of total messages in the network be denoted as $K(t)$, and the buffer capacity of each node be denoted as C. messages. The messages are generated arbitrarily between source and destination nodes. Each message is destined to one of the nodes in the network with a time-to-live (denoted as Tx). A message is dropped if its Tx expires.

For any given node, A, it is assumed that $J_A(t)$ messages are stored in its buffer at time t. Each message i, $i \in [1, J_A(t)]$ is denoted by a tuple of variables denoted in Table 6.1. Obviously we have $s_i(t) = n_i(t)$ if all the encountered nodes of message i have available buffer space, and $n_i(t) \leq m_i(t) + 1$. Let the expected inter-encounter time of any two nodes A and B be denoted as $E[\triangle T_{(A,B)}]$, which is defined as the average time period taken by the two nodes to enter into their transmission again. The encounter (or mixing) rate between A and B, denoted as $\beta_{(A,B)}$, is the inverse of the expected inter-encounter time for the two nodes: $\beta_{(A,B)} = \frac{1}{E[\triangle T_{(A,B)}]}$. We assume that $E[\triangle T_{(A,B)}]$, A, $B \in [1, N]$ follows an exponential distribution (or referred to as with an exponential tail [45]). It has been shown that a number of popular mobility models have such exponential tails (e.g., Random Walk, Random Waypoint, Random Direction, Community-based Mobility [50, 5]). Recent studies based on traces collected from real-life mobility examples [66] argued that the inter-encounter period and the encounter durations in these traces demonstrate exponential tails after a specific cut-off point. Based on the *iid* of the nodal mobility model, the distribution of the inter-meeting time can be obtained, where the historical inter-encounter information between any two nodes A and B can be calculated by averaging all inter-encounter times until current time t. This distribution is common for all nodes in the network. Thus, the parameter of the exponential distribution, denoted as β can be expressed as :

$$\beta = \frac{1}{E[\triangle T_{(A,B)}]} \approx \frac{1}{\frac{1}{n}\sum_{k=1}^{n}\triangle T_{(A,B)}^{(k)}} \tag{6.8}$$

Where n is the number of encounters until current time t. The adaptation of the mobility characteristics becomes more precise with a greater elapsed time as the historical information becomes more viable.

6.1.4 Proposed Buffer Management and Scheduling Framework

Figure 6.1 provides the whole picture on the proposed DTN buffer management framework, which illustrates the functional modules and their relations. The summary vector exchange module (SVEM) is implemented at a node during a contact; then the network state estimation module (NSEM) is used to estimate the values

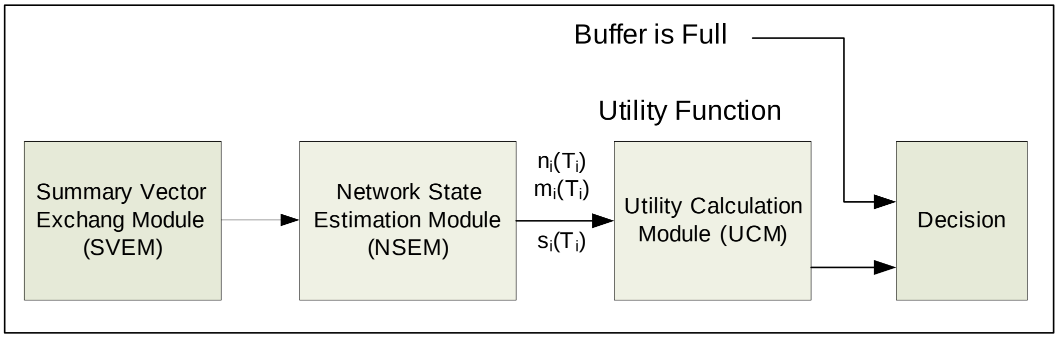


Figure 6.1: The buffer management framework

of $m_i(T_i)$, $n_i(T_i)$, and $s_i(T_i)$ according to the most updated network information. The two parameters are further taken as inputs in the calculation of the proposed per-message utility function in the utility calculation module (UCM). The decision of forwarding or dropping the buffered messages is made based on the buffer occupancy status and the utility value of the messages. The rest of the section introduces the details of each module.

Summary Vector Exchange Module (SVEM)

During each contact, the network information summarized as a "summary vector", is exchanged between the two nodes, which includes the following data: (1) statistics of inter-encounter time of every node pair maintained by the nodes, (2) statistics regarding the buffered messages, including their IDs, remaining time to live (R_i), destinations, the stored $n_i(T_i)$, $m_i(T_i)$, and $s_i(T_i)$ values for each message that were estimated in the previous contact. The SVEM ensures the above information exchange process, and activates NSEM for the parameter estimation based on the newly obtained network statistics right after each contact.

Proposed Network State Estimation Module (NSEM)

The NSEM is used to obtain the estimated $m_i(T_i)$, $n_i(T_i)$, and $s_i(T_i)$ such that the UCM can make decision in the buffer management process. Since acquiring global information about a specific message may take a long time to propagate and hence might be obsolete when we calculate the utility function of the message, we come up with a time-window based estimation approach. Rather than using the current value of $m_i(T_i)$ and $n_i(T_i)$ for a specific message i at an elapsed time T_i, we use the measure of the two parameters over the messages that node A is aware of (has "seen") during an elapsed time T_i. These estimations are then used in the evaluation of the per-message utility.

For this purpose, we propose a novel estimation approach called Global History-Based Prediction (GHP), which estimates the parameters by considering their statistics since the corresponding message was created. Let $M_i(T_i)$, $N_i(T_i)$, and $S_i(T_i)$ denote random variables that fully describe the parameters $m_i(T_i)$, $n_i(T_i)$, and $s_i(T_i)$ at elapsed time T_i, respectively. We have: $E[M_i(T_i)] = \frac{\sum_{i=1}^{j} m_i(T_i)}{j}$, $E[N_i] = \frac{\sum_{i=1}^{j} n_i(T_i)}{j}$, and $E[S_i] = \frac{\sum_{i=1}^{j} s_i(T_i)}{j}$, where j is the total number of messages that have been seen by node A and generated before message i. These messages include the messages stored in the buffer of A that are considered more senior than message i. In the same manner, the average elapsed times for all messages that were generated before message i is calculated as $\widehat{T} = \frac{\sum_{i=1}^{j} T_i}{j}$. Thus, we can have the following estimations for message i: $\widehat{m_i(T_i)}$, $\widehat{n_i(T_i)}$ and $\widehat{s_i(T_i)}$. These values are then incorporated into the per-message utility metrics, which are calculated as $\widehat{m_i(T_i)} = \frac{T_i E[M_i]}{\widehat{T}}$, $\widehat{n_i(T_i)} = \frac{T_i E[N_i]}{\widehat{T}}$ and $\widehat{s_i(T_i)} = \frac{T_i E[S_i]}{\widehat{T}}$.

Utility Calculation Module (UCM)

Based on the problem settings and estimated parameters, the UCM answers the following question at a node during each nodal contact: Given $n_i(T_i)$, $m_i(T_i)$, $s_i(T_i)$ and limited buffer space for supporting epidemic or source forwarding routing [29, 5], what is an appropriate decision on whether the node should drop any message in its buffer or reject any incoming message from the other node during the contact, such that either the average delivery ratio or delivery delay can be optimized? Section 5 describes in details how the per-message utility function is derived. The scenarios of interest in the study are as following:

- Maximizing the delivery ratio under epidemic routing
- Maximizing the delivery ratio under source forwarding
- Minimizing the average delivery delay under epidemic routing
- Minimizing the average delivery delay under source forwarding

6.1.5 Forwarding and Dropping Policy

With the per-message utility, the node firstly sorts the buffer messages accordingly from the highest to the lowest. The messages with lower utility values have higher priorities to be dropped when the node's buffer is full, while the messages with higher utility values have higher priorities to be forwarded to the encountered node. Figure 6.2 illustrates the forwarding and dropping actions: if the utility U_j of message j (the message with the highest utility value) buffered in A is higher than U_C of message i (the message with the lowest utility value) at node B, then message i is dropped and replaced by a copy of message j if the buffer of B is full during the contact of the two nodes.

6.1.6 Utility Function Derivation

Maximization of Delivery Ratio

Let us assume that the buffer is full at node B and there is a message i with elapsed time T_i in a network that has K messages at the moment at which the decision should be made by a node with respect to dropping a message from all messages in its buffer.

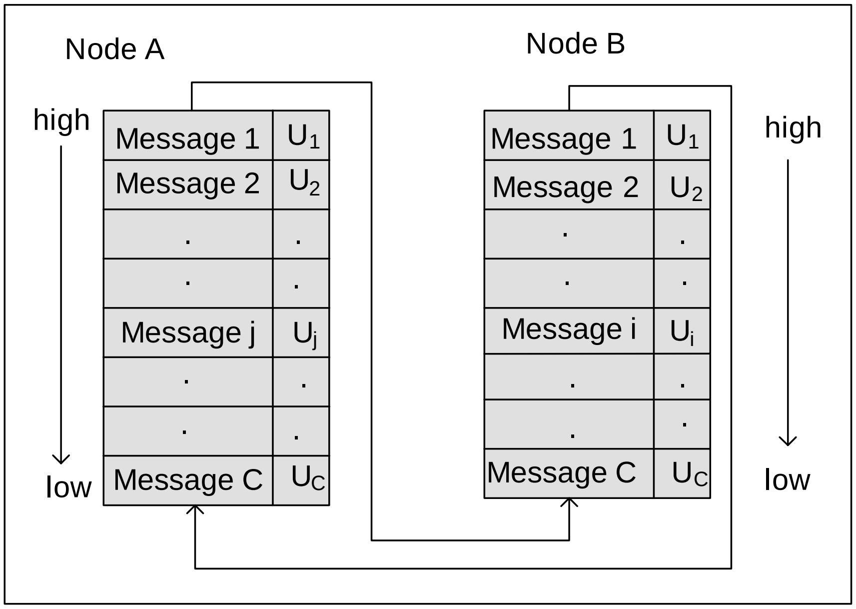


Figure 6.2: The forwarding and dropping at a node

Epidemic Forwarding

To maximize the average ratio of all messages, a node should therefore drop message i_{min} that satisfies the following:

$$i_{min} = argmin_i \left[\left(1 - \frac{m_i(T_i)}{N-1}\right)^2 * \right.$$

$$\left(\frac{N}{N - n_i(T_i) + n_i(T_i).e^{\beta N R_i P_{f_i}}}\right)^{m_i(T_i)+1}$$

$$\left. * \left[e^{\beta N R_i P_{fi}} \left(\beta R_i n_i(T_i) + \frac{m_i(T_i)}{N}\right) - \frac{m_i(T_i)}{N} \right] \right] \qquad (6.9)$$

where P_{fi} is the probability of forwarding message i to every encountered node which can be estimated as $P_{fi} = \frac{n_i(t)}{m_i(t)}$.

Proof: The probability that a copy of message i will not be delivered by a node is given by the probability that the next meeting time with the destination is greater than its remaining lifetime R_i , assuming that the message i has not yet been delivered. The probability that message i will not be delivered (i.e., none of its copies will be delivered) can be expressed as

$$Pr\{message\, i\, not\, delivered \,|\, not\, delivered\, yet\} =$$

$$P(T_d < T_i + R_i \,|_{T_d > T_i}) = 1 - \left[\left(1 - \frac{m_i(T_i)}{N-1}\right)\right.$$

$$\left.* \left(\frac{N}{N - n_i(T_i) + n_i(T_i)e^{\beta P_{fi} N R_i}}\right)^{\frac{n_i(T_i)}{P_{f_i}}}\right] \tag{6.10}$$

The proof of (6.10) is provided in Appendix.

By assuming network homogeneity, there is an equal likelihood that the message is "seen" by each node. Thus, the probability that message i has been already delivered to the destination is equal to

$$Pr\{\, message\, i\, already\, delivered\,\} = \frac{m_i(T_i)}{N-1} \tag{6.11}$$

By combining (6.10) and (6.11), the probability that message i is successfully delivered before its Tx expires can be calculated as follows:

$$Pr_i = 1 - P\{message\, i\, not\, yet\, delivered\}$$

$$*P\{message\, i\, will\, not\, be\, delivered\, within\, R_i\}$$

$$Pr_i = 1 - \left(1 - \tfrac{m_i(T_i)}{N-1}\right)^2$$

$$* \left(\frac{N}{N - n_i(T_i) + n_i(T_i).e^{\beta P_{fi} N R_i}} \right)^{m_i(T_i)} \tag{6.12}$$

When a node is operating at its maximum buffer capacity, it should drop one or multiple messages so as to achieve the best gain in the increase of the global delivery ratio $Pr = \frac{1}{K(t)} \sum_{i=1}^{K(t)} Pr_i$. To make the optimal decision locally at the node, Pr_i is differentiated with respect to $n_i(T_i)$, and $\partial n_i(T_i)$ is then discretized and replaced by $\triangle n_i(T_i)$.

The best drop policy is one that maximizes $\triangle Pr_i$:

$$\begin{aligned}
&\triangle Pr_i = \tfrac{\partial Pr_i}{\partial n_i(T_i)} * \triangle n_i(T_i) \\
&= \left[\left[e^{\beta N R_i P_{f_i}} (\beta R_i n_i(T_i) + \tfrac{m_i(T_i)}{N}) - \tfrac{m(T_i)}{N} \right] \left(1 - \tfrac{m_i(T_i)}{N-1} \right)^2 \right. \\
&\left. * \left(\tfrac{N}{N - n_i(T_i) + n_i(T_i) e^{\beta N R_i P_{fi}}} \right)^{m_i(T_i)+1} \right] \triangle n_i(T_i)
\end{aligned}$$

Thus, the maximum delivery ratio can be achieved if the message that causes the least decrease in $\triangle Pr_i$ is discarded. On the other hand, when message i is discarded, the number of copies of message i in the network decreases by 1, which results in $\triangle n_i(T_i) = -1$. Thus the optimal buffer dropping policy that can maximize the delivery ratio based on the locally available information at the node is to discard the message with the smallest value of $\frac{\partial Pr_i}{\partial n_i(T_i)}$, which is equivalently to choose a message with a value for i_{min} that satisfies (6.9). This derivation is an attempt to handle changes in the number of copies of a message that may be increased in the future during new encounters. This goal can be achieved by predicting P_f, the probability of forwarding a copy of message i to any node encountered, which is incorporated into the estimation of the delivery ratio. It is clear that the accuracy of P_f is based mainly on the precision in estimating the values of $m_i(T_i)$ and $n_i(T_i)$.

Source Forwarding

Since only L message copies are allowed to be spread by the source node, it is very important to estimate the time at which the L message copies have been spread in the network, which is denoted as T_{Li}. Whether the value of T_i is less or greater than T_{Li} plays key role in formulating the utility function. To simplify the notation , we use term T_{Ri} for the period $T_{Li} - T_i$, and T_{XLi} for $Tx - T_{Li}$.

The local optimal buffer management policy that maximizes the average delivery ratio is to drop message i_{min} that satisfies the following:

$$argmin_i =$$

$$\begin{cases} \left[\frac{1}{Pf_i}\left(1-e^{-\beta P_{fi} T_{Ri}}\right)\left(1-\frac{m_i(T_i)}{N-1}\right)^2 e^{\beta N T_{Ri}} \right. \\ \left. e^{\frac{1}{P_{f_i}}(N-n(T_i))(e^{-\beta P_{fi} T_{Ri}}-1)}\right] e^{-\beta L (T_{XLi})},\ T_i < T_{Li} \\ \\ \beta P_{fi} R_i \left(1-\frac{m_i(T_i)}{N-1}\right)^2 e^{-\beta P_{fi} L R_i}, \quad T_i \geq T_{Li} \end{cases} \tag{6.13}$$

P_{fi} under source forwarding is estimated as $P_{fi} = \frac{s_i(T_i)}{m_i(T_i)}$. Note that the estimation of P_{fi} is different from that of epidemic forwarding since we deal with a controlled flooding case.

Proof: The probability that message i will be delivered (i.e., that n_i copies are delivered) within the remaining lifetime of the message can be expressed by

$$Pr_i\{message\, i\, will\, be\, delivered\, within\, R_i\} =$$

$$P(T_d \leq T_i + R_i\ |_{T_d > T_i}) =$$

$$
= \begin{cases} 1 - \left[(1 - \frac{m_i(T_i)}{N-1}) e^{\beta N T_{Ri} + \frac{1}{P_{fi}}(N - n(T_i))} \right. \\ \left. e^{(e^{-\beta P_{fi} T_{Ri}} - 1)} \right] e^{-\beta L(T_{XLi})}, \; T_i < T_{Li} \\ \\ \\ 1 - \left(1 - \frac{m_i(T_i)}{N-1}\right) e^{-\beta P_{fi} L R_i}, \; T_i \geq T_{Li} \end{cases} \tag{6.14}
$$

The proof of equation (6.14) is included in Appendix.

Since the node's mobility is *iid*, the probability that message i has been already delivered is equal to

$$
Pr\{ \, message \, i \, already \, delivered \, \} = \frac{m_i(T_i)}{N-1} \tag{6.15}
$$

When (6.14) and (6.15) are combined, the probability that message i will be delivered before its Tx expires is given by the total probability law as

$$
Pr_i = 1 - P\{message \, i \, not \, yet \, delivered\} *
$$

$$
P\{message \, i \, will \, not \, be \, delivered \, within \, R_i\}
$$

$$
Pr_i = \begin{cases} 1 - \left[\left(1 - \frac{m_i(T_i)}{N-1}\right)^2 e^{\beta N T_{Ri} + \frac{1}{P_{fi}}(N - n(T_i))} \right. \\ \left. e^{(e^{-\beta P_{fi} T_{Ri}} - 1)} \right] e^{-\beta L(T_{XLi})}, \; T_i < T_{Li} \\ \\ \\ 1 - \left(1 - \frac{m_i(T_i)}{N-1})\right)^2 e^{-\beta P_{fi} L R_i}, \; T_i \geq T_{Li} \end{cases} \tag{6.16}
$$

In the case of congestion, a DTN node should drop the message that leads to the best gain in the global delivery ratio. To find the local optimal decision, Pr_i is differentiated with respect to $n(T_i)$ if $T_i < T_{Li}$ and to L otherwise, and ∂n is then discretized and replaced by $\triangle n$.

$$\triangle Pr_i = \frac{\partial Pr_i}{\partial n} * \triangle n(T_i) =$$

$$\begin{cases} \left[\frac{1}{Pf}\left(1 - e^{-\beta P_{fi} T_{Ri}}\right)\left(1 - \frac{m_i(T_i)}{N-1}\right)^2 e^{\beta N T_{Ri}} \\ e^{\left(e^{-\beta P_{f_i} T_{Ri}} - 1\right) + \frac{1}{P_{f_i}}(N - n(T_i))}\right] e^{-\beta L(T_{XLi})}, \ T_i < T_{Li} \\ \\ \beta P_{f_i} R_i \left(1 - \frac{m_i(T_i)}{N-1}\right)^2 e^{-\beta P_{fi} L R_i}, \ T_i \geq T_{Li} \end{cases}$$

After the $T_i \geq T_L$, the number of message copies will be subject to be decreased due to discarding the message that has the highest number of message copies. Therefore the second part can be differentiated with respect to L:

$$\frac{\partial P}{L} = \beta P_{f_i} R_i \left(1 - \frac{m_i(T_i)}{N-1}\right) e^{-\beta P_{fi} L R_i}$$

The optimal buffer dropping policy that maximizes the probability of delivery is thus to discard the message that has the smallest value of $\frac{\partial Pr_i}{\partial n(T_i)}$, that is to choose a message with a value for i_{min} that satisfies (6.13).

Minimization of Average Delivery Delay

To minimize the average delivery delay, node A should discard a message such that the expected delivery delay of all messages can be reduced the most. Since the delivery delay of the messages is mainly affected by the nodal inter-encounter time, we assume that all message have infinite or large enough Tx and derive the utility function such that it is affected by number $m_i(T_i)$, $n_i(T_i)$, P_{fi}, and enter-encounter time.

Epidemic Forwarding

To achieve the minimum average delivery delay, node A should drop the message that satisfies the following:

$$i_{min} = \left(1 - \frac{m_i(T_i)}{N-1}\right) *$$

$$\left[\frac{Ln(N).m(T_i)\left[N - 2n(T_i)\right]}{\left(Nn(T_i) - n_i^2(T_i)\right)^2 \beta}\right] \tag{6.17}$$

Proof: The expected delay in delivering a message that still has copies existing in the network can be expressed

$D_i = P\{message\, i\, not\, deliverd\, yet\} * \frac{1}{P_{fi}} E[T_d \mid T_d > T_i]$

$$D_i = \left(1 - \frac{m_i(T_i)}{N-1}\right) * \frac{1}{P_{fi}} E[T_d \mid T_d > T_i] \tag{6.18}$$

where

$$E[T_d \mid T_d > T_i] = \left[T_i + \frac{ln(N)}{P_{fi}\beta(N - n(T_i))}\right] \tag{6.19}$$

Proof: The proof of (6.18) is provided in Appendix.

$$D_i = \left(1 - \frac{m_i(T_i)}{N-1}\right) \left[T_i + \frac{m(T_i)ln(N)}{\beta(Nn(T_i) - n^2(T_i))}\right] \tag{6.20}$$

When a node buffer is full, the node should make a drop decision that leads to the least increase on D_i. To find the local optimal decision, D_i is differentiated with respect to $n_i(T_i)$, and ∂D_i is then discritized and replaced by $\triangle D_i$:

$$\triangle D_i = \frac{\partial D_i}{\partial n_i(T_i)} * \triangle n_i(T_i)$$

$$= \left(1 - \frac{m_i(T_i)}{N-1}\right) \left[\frac{-ln(N).m(T_i)\left[N - 2n(T_i)\right]}{\left(Nn(T_i) - n_i^2(T_i)\right)^2 \beta} \right] \triangle n_i(T_i)$$

To reduce the delivery delay of all messages existing in the network, the best decision is to discard the message that maximizes the total average of the delivery delay, $D = \frac{1}{K(t)} \sum_{i=1}^{K(t)} D_i$, among all the messages. Therefore, the optimal buffer-dropping policy that maximizes the delivery delay is thus to discard the message that has the min value of $\mid \frac{\partial D_i}{\partial n_i(T_i)} \mid$, which is equivalently to choose a message with a value for i_{min} that satisfies (6.17).

Source Forwarding

To minimize the delivery delay of all messages, node i should discard the message that increases the expected delivery delay of all messages. To minimize the average delay of all messages, a node should therefore drop message i_{min} that satisfies the following:

$$argmin_i =$$

$$\begin{cases} \frac{1}{P_f}\left(1 - \frac{m(T_i)}{N-1}\right)^2 \left[\frac{1}{(n(T_i)+1)^2\beta}\right], T_i < T_{Li} \\ \\ \frac{1}{P_{f_i}}\left(1 - \frac{m_i(T_i)}{N-1}\right) \left[\frac{1}{P_{f_i} L^2(T_i)\beta}\right], T_i \geq T_{Li} \end{cases} \tag{6.21}$$

Proof: The expected delay in delivering a message that still has copies existing in the network is

$$D_i = P\{message\, i\, not\, deliverd\, yet\} * \frac{1}{P_{fi}} E[T_d \mid T_d > T_i]$$

$$D_i = \begin{cases} \left(1 - \frac{m_i(T_i)}{N-1}\right) * \left[T_i + \frac{1}{P_{f_i}}\left(1 - \frac{m(T_i)}{N-1}\right) * \right. \\ \qquad \left. \frac{1}{(n(T_i)+1)\beta}\right],\ T_i < T_{Li} \\ \\ \left(1 - \frac{m_i(T_i)}{N-1}\right)\left[T_i + \frac{1}{P_{f_i} L\beta}\right], \quad T \geq T_{Li} \end{cases} \tag{6.22}$$

The proof of (6.22) is included in Appendix.

When a node buffer is full, a node should make a drop decision that leads to the largest decreasing on D_i. To find the local optimal decision, D_i is differentiated with respect to $n(T_i)$ if $T_i < T_{Li}$ and with respect to L otherwise, and ∂D_i is then discritized and replaced by $\triangle D_i$.

$$\triangle D_i = \frac{\partial D_i}{\partial(n(t)/L)} * \triangle(n(T_i)/L)$$

To reduce the delivery delay of all messages in the network, the best decision is to discard the message that maximizes the total delivery delay, $\mid \triangle D \mid$, of all messages. Therefore, the optimal buffer-dropping policy that maximizes the delivery delay is thus to discard the message that has the minimum value of $\mid \frac{\partial D_i}{\partial(n(t)/L)} \mid$, that is to choose a message with a value for $i_{min} = argmin \triangle D_i$ that satisfies equation (6.21). This policy drops a message that has the highest number of message copies within shortest elapsed time since the creation of the message.

6.1.7 Performance Evaluation

Experimental Setup

To examine the efficiency of the proposed buffer management architecture [82], experiments were conducted, and the results presented in this section. To better understand the performance of the proposed estimation strategy–GHP, we also implement two other estimation strategies for the values of $m_i(T_i)$, $n_i(T_i)$, and $s_i(T_i)$ namely Global Knowledge-based Management (GKM), and Encounter History-Based Prediction (EHP).

The GKM assumes knowing the exact values of $m_i(T_i)$, $n_i(T_i)$, and $s_i(T_i)$ and is supposed to achieve the best performance. Since such an assumption is not practical [80], the result of GKM is taken as a benchmark for the proposed GHP scheme. With EHP, two encountered nodes update each other with respect to the messages they have in common, and the values of $m_i(T_i)$, $n_i(T_i)$,and $s_i(T_i)$ are updated accordingly. This policy of update provides a sub-optimal solution and has been employed in [69] and [76]. In addition to the above prediction strategies, we compared the proposed buffer management architecture with three well-known policies listed as follows:

- History-based drop (HBD) [76] is based on the history of all messages (on average) in the network after an elapsed time. The variables of the message utility are estimated by averaging the variables of all messages in the network after the elapsed time.
- Drop oldest (DO) drops the message with the shortest remaining time to live.
- Drop front (DF) drops the message that entered the queue the earliest when the buffer is full. This policy obtains the best performance of all the policies used by Lindgren et al. in [77].

We assume a message issued at a node (termed sourced messages) has the highest priority at the node. If all buffered messages are sourced ones and the newly arrived message is also a source message at the node, then the oldest one is dropped.

This idea was examined in [33] and has been proved with improved delivery ratio. To evaluate the policies, a DTN simulator similar to that in [50] is implemented. The simulations are based on two mobility scenarios; a synthetic one based on Random Waypoint mobility model, and a real trace-like mobility model based on a real traces of Zebranet experiment. The real trace was collected as part of the ZebraNet wildlife tracking experiment described in [87, 92, 6]. The mobility under this model is constructed from distributions that match the traces collected from real movements of zebras. The speed and the turning angle selection process are repeated for the whole experimental study duration. The simulation parameters are as shown in table 6.2. Each node has a transmission range, $D = 30$ meters, to obtain sparsely populated network. Euclidean distance is used to measure the proximity between two nodes (or their positions). A slotted collision avoidance MAC protocol with Clear-to-Send (CTS) and Request-to-Send (RTS) features was implemented in order to arbitrate between nodes that contend for a shared channel. The message inter-arrival time is uniformly distributed in such a way that the traffic can be varied from low (10 messages generated per node) to high (70 messages generated per node). The buffer size is set to a low capacity (15 messages), to push the network towards a congestion state by increasing the network traffic. We assume sufficient time for completing the possible message exchange for every contact. Message delivery ratio and the delivery delay are taken as two performance measures. Each data is the average of the results from 30 runs. A PC with Intel 2.0 Ghz Core 2 Duo processor and 2 GB RAM is used for running the simulations.

Table 6.2: Simulation parameters of RWP and ZebraNet mobility scenarios

Mobility Pattern	RWP	ZebraNet
Simulation Duration (hours)	11	15
Simulation Area	800×800	1000×1000
No. of Nodes	130	80
Average Speed (m/s)	3	-
TTL (hours)	2	3

Introduced Policy for Maximizing Delivery Ratio

This section examines the proposed policy for maximizing the average delivery ratio under the considered scenarios. The plots of the delivery ratio obtained for epidemic and source forwarding under random Waypoint mobility model is shown in Figure 6.3.

It can be seen that GKM gives the best performance under all the traffic loads for both routing schemes due to the complete and global mobility information. The GHP policy provides the next best result and is competitive with the GKM in the case of low traffic. As the traffic increases, the performance of all policies degrades, while the GHP outperforms all other policies except GKM. For epidemic routing, it can achieve a delivery ratio up to 215% higher than that achieved by DO, 70% higher than DF, 22% higher than HBD, 32% higher than EHP, and only 15% worse than GKM. For source forwarding, GHP provides results can be up to 200% higher than DO, 80% higher than DF, 17% higher than HBD, 30% higher than EHP, and 11% worse than GKM. It is noted that all policies obtain better results for source forwarding than that for epidemic routing, which proved that the controlled flooding mechanism improves performance in the case of limited buffer capacity. Figure 6.4 shows the results of delivery ratios under Zebranet trace. As can be seen, GHP can achieve a delivery ratio 249% higher than that achieved by DO, 93% higher than DF, 16% higher than HBD, 29% higher than EHP, and only

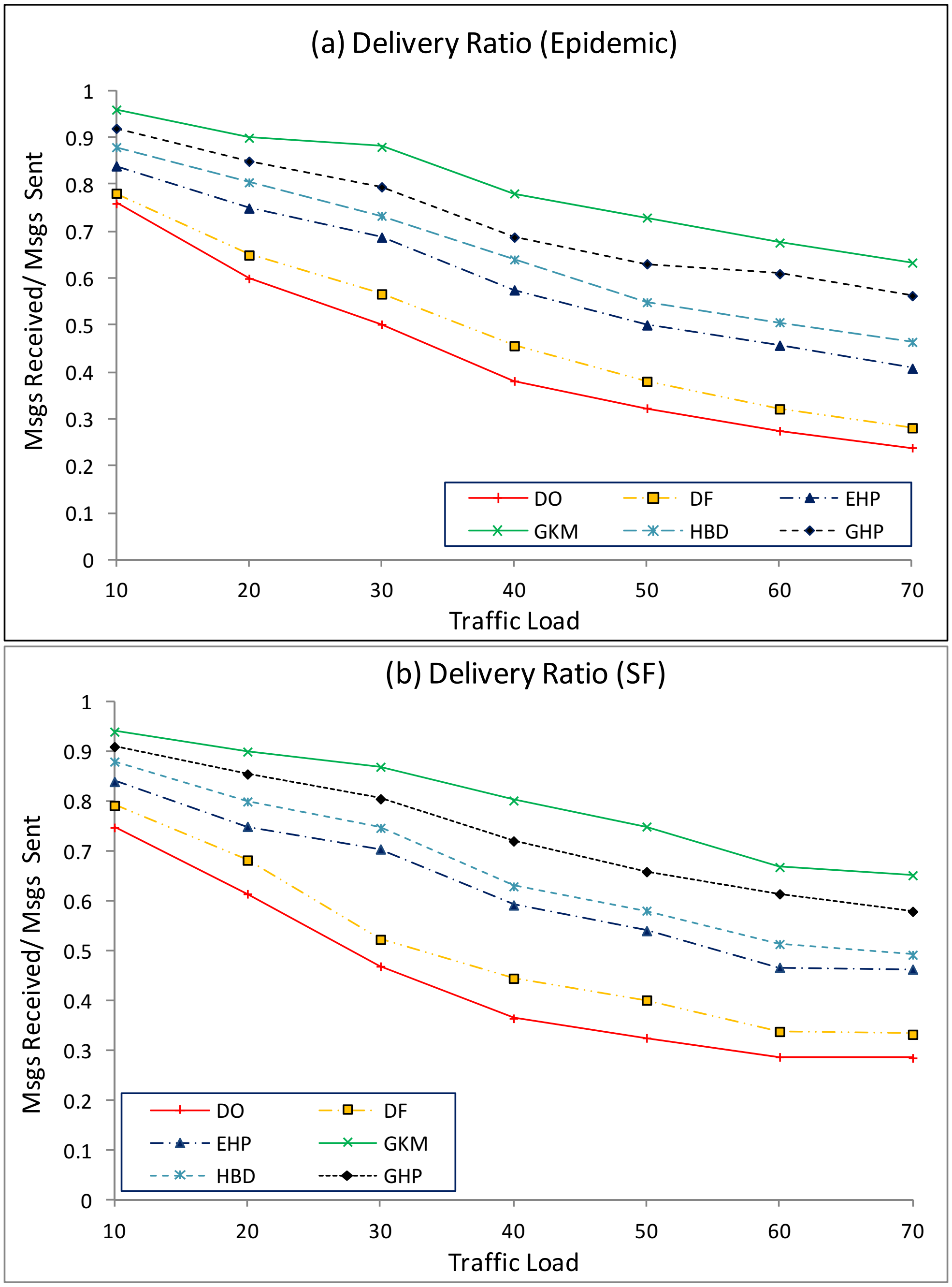


Figure 6.3: The effect of traffic load on the delivery ratio under Random Waypoint mobility model

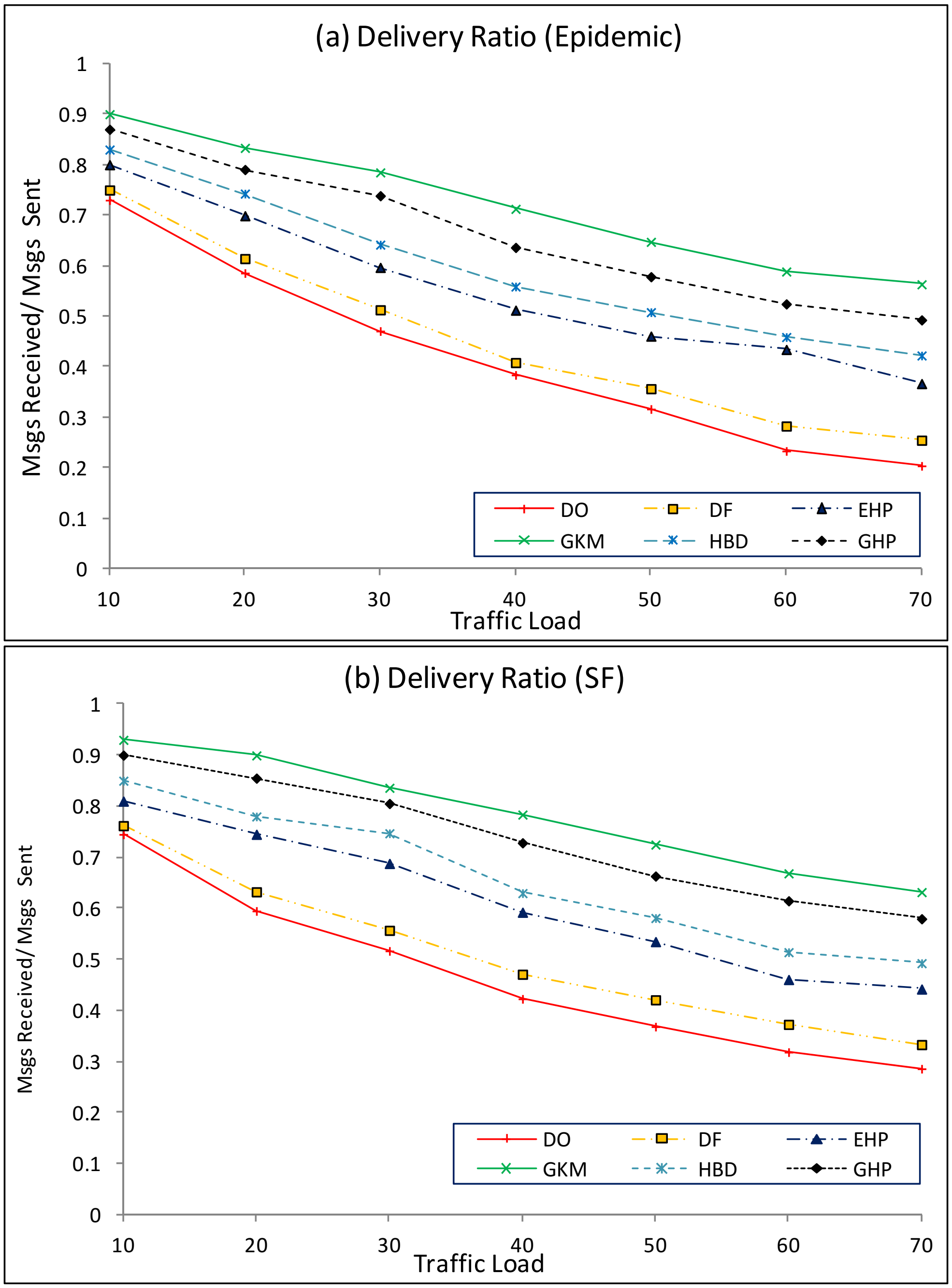


Figure 6.4: The effect of traffic load on the delivery ratio under ZebraNet trace

13% worse than GKM. For source forwarding, GHP provides results can be 210% higher than DO, 73% higher than DF, 15% higher than HBD, 28% higher than EHP, and 9% worse than GKM.

Introduced Policy for Minimizing Delivery Delay

This subsection evaluates the effect of the policy of each routing scheme on message delivery delay using the same scenarios in the previous section.

Figure 6.5 shows the results. Similarly, the GKM gives the best performance under all traffic loads for both routing techniques, while the GHP is the second best and is competitive with the GKM in the case of low traffic. As the traffic increases, the demand on the wireless channel and buffers increases, causing a long queuing delays and substantial message loss that negatively affect the performance of all the examined policies. We have observed that for both routing schemes the GHP outperforms all other policies. GHP under epidemic routing is better than DO by 42%, DF by 53%, HBD by 10%, EHP by 20%, and a longer delay of only 8% of that achieved by GKM. Under source forwarding, GHP can reach delivery delays up to 57% shorter than DF, 44% shorter than DO, 17% shorter than HBD, 27% shorter than EHD, and only 10% longer than GKM. Figure 6. shows the results of delivery delay under ZebraNet trace. As can be seen, GHP under epidemic routing is better than DF by 81%,DO by 71%, HBD by 15%, EHP by 24%, and a longer delay of only 11% of that achieved by GKM. Under source forwarding, GHP can reach delivery delays up to 66% shorter than DF, 53% shorter than DO, 14% shorter than HBD, 22% shorter than EHD, and only 12% longer than GKM.

6.1.8 Computation and Performance Tradeoff

It is clear that GHP outperforms their counterparts thanks to the adaptive calculation of utility values using a couple of global network state parameters, i.e., $m(T)$

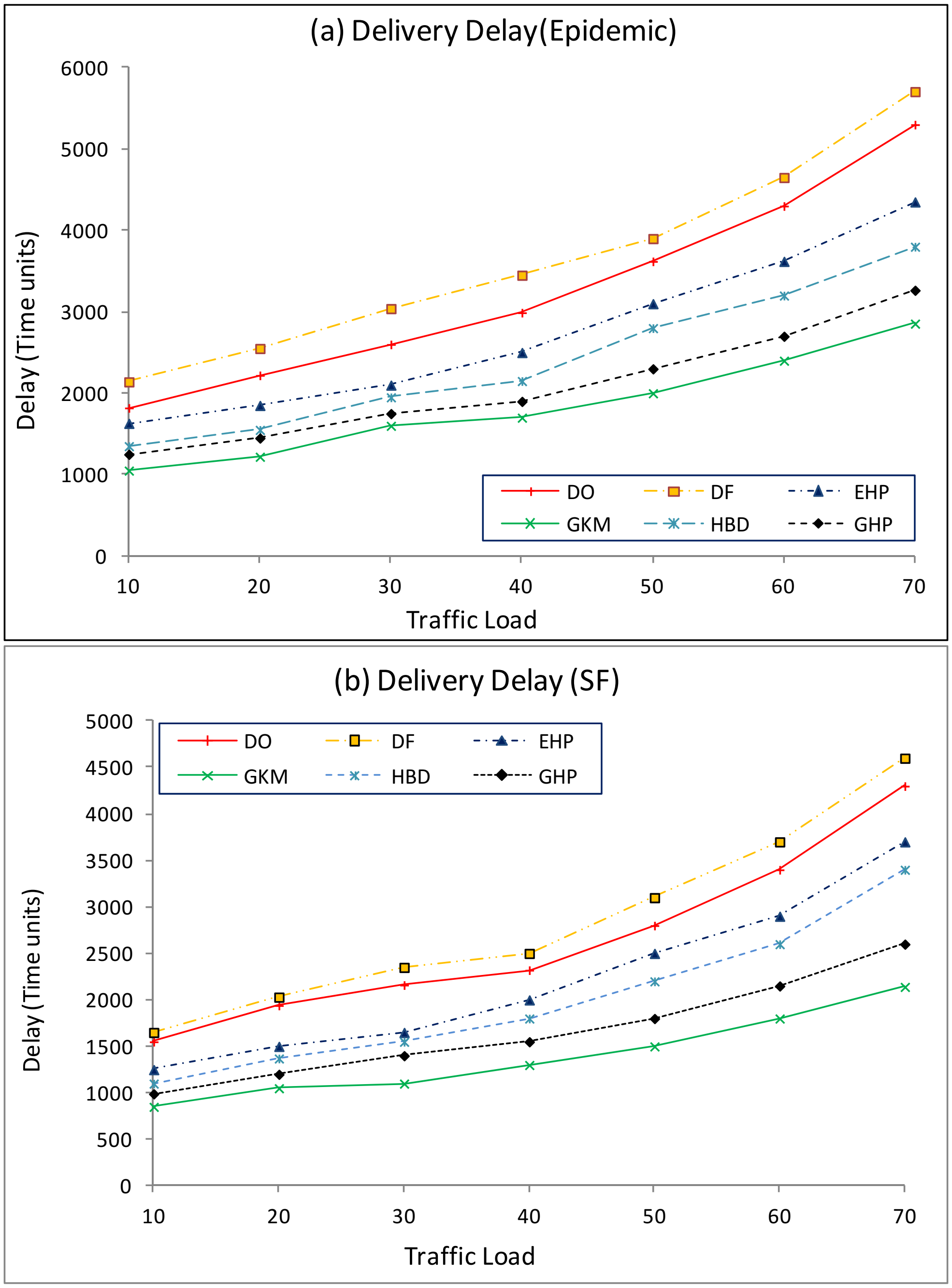


Figure 6.5: The effect of traffic load on delivery delay under Random Waypoint Mobility

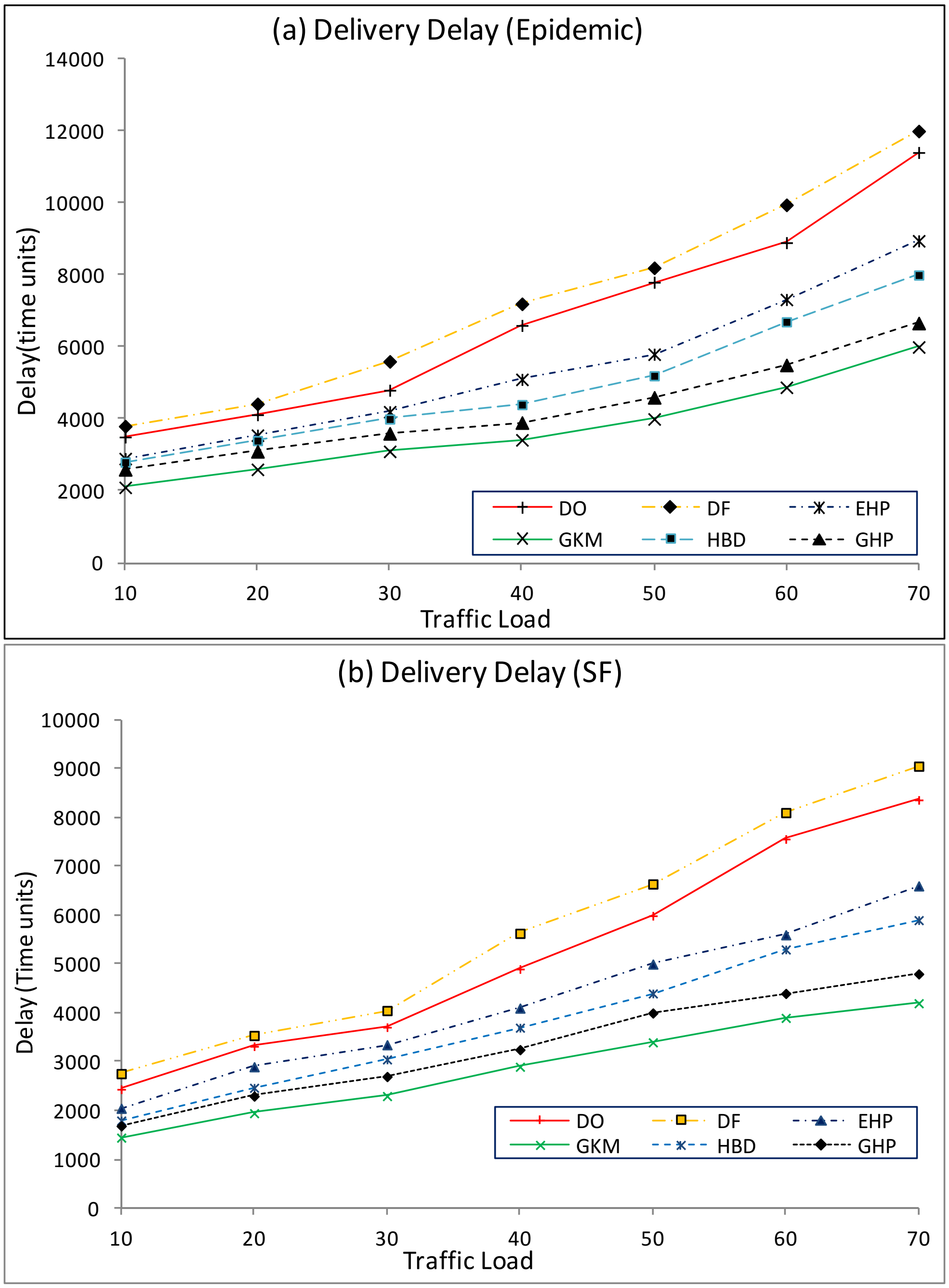


Figure 6.6: The effect of traffic load on delivery delay under ZebraNet trace

and $n(T)$. Such performance gain, nonetheless, is at the expense of larger computation complexity, which in turn causes longer computation time. We evaluate the proposed message scheduling approach as follows in terms of the impact due to computation complexity.

It is clear that the main source of computation complexity lies in calculating the utility function, which is in turn dominated by the number of messages involved in the solution process. Another fact is that, considering more messages is expected to yield better precision in the utility function calculation (so as for the overall performance) at the expense of longer computation time. Let *Sampling List* (SL) be the subset of randomly selected messages for consideration in the utility function calculation at a node. The size of SL stands for the amount of statistics collected at the node to make the message forwarding/dropping decision. Our strategy is to get the relation between the performance and the size of subset, and then the relation between the size of subset and the computation time. Thus we will be able to observe the performance gain due to the longer computation time compared with its counterparts.

To examine the desired scenario with high congestion, the buffer size is set to 20, and traffic load is set high (90 generated message per node). Without loss of generality, this scenario is performed under epidemic routing using Random Waypoint mobility model.

The performance impact on GHP by reducing the amount of collected statistics is shown in Figure 6.7.

It is shown that increasing the SL size in GHP results in the corresponding performance improvement over EHP.

When the SL size is 1, GHP is degraded as EHP since it becomes completely based on local information. As the SL size is increased, the performance of GHP improves considerably. Figure 6.8 shows the relation between the computation

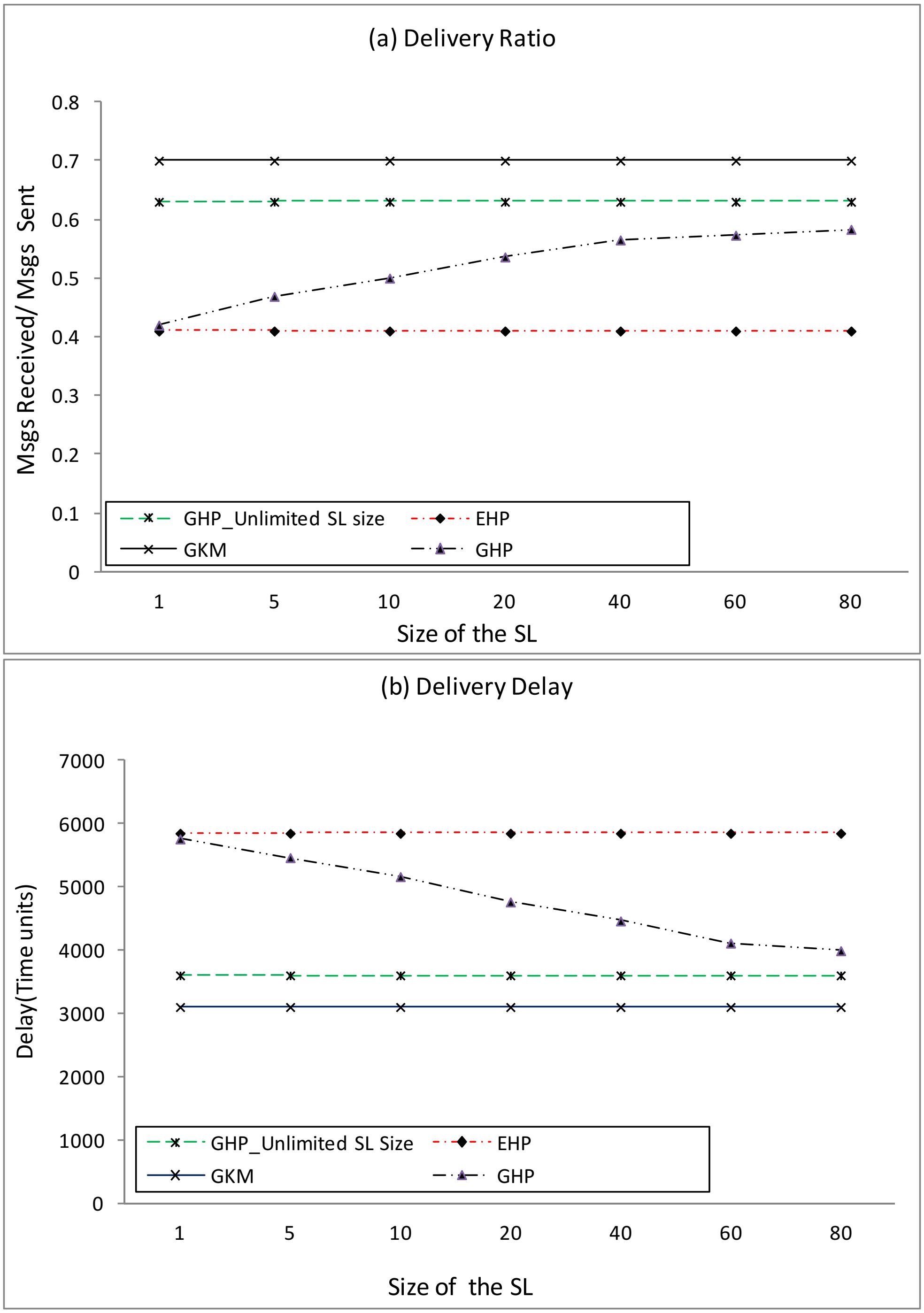


Figure 6.7: Impact of SL size on the delivery ratio and delivery delay

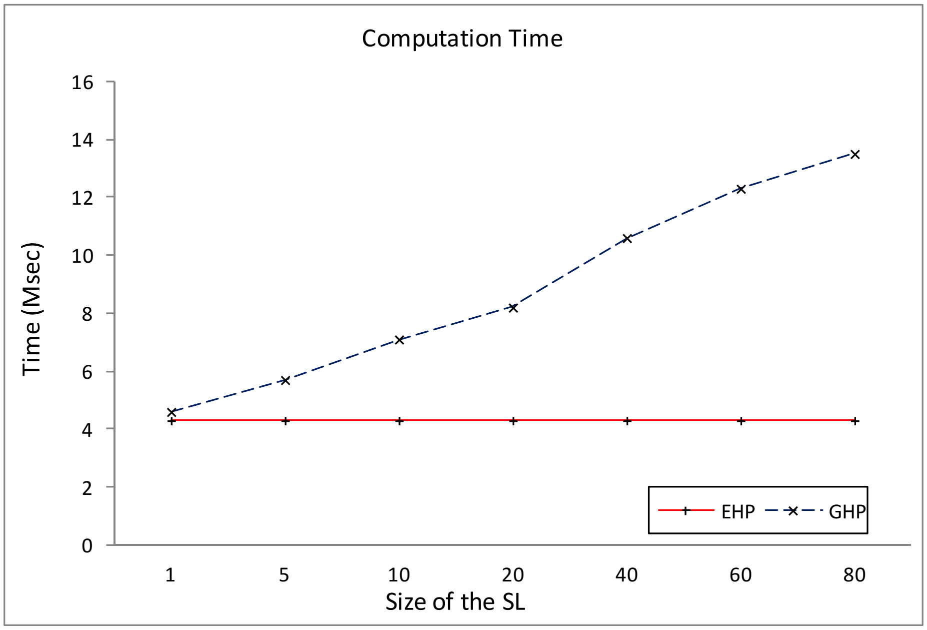


Figure 6.8: The effect of SL size on the policy computation time

complexity and the SL size.

Note that HBD and GHP under a unlimited SL size yield 6 and 7.5 times of longer computation time than that by GHP at SL size of 40 messages, respectively, which are not shown in the chart. The very long computation time is due to the fact that all the messages that a node has been learned from all encountered nodes are considered in the calculation of the utility function.

Our results suggest that, with a carefully designed statistics collection strategy, the proposed GHP scheme can be manipulated to achieve a graceful tradeoff among the computation time (which is directly related to nodal power consumption) and performance according to any desired target function. In addition, we have seen that GHP only takes a small fraction of computation time compared to the case by maintaining a complete view on all the messages older than message i, while without significantly affecting the performance.

6.2 Buffer Management and Scheduling for Utility-based Routing

Although SAURP and ARBRP have improved the previously reported designs in terms of the message delivery delay and the delivery ratio, they are still subject to respective problems due to limited network resources such as buffer space. Thus, to achieve efficient utilization of network resources, it is important to come up with an effective message scheduling strategy to determine which messages should be forwarded and which should be dropped in case of buffer is full. This section investigates a new buffer management and scheduling framework for SAURP [85]. The decision of forwarding or dropping the buffered messages is made based on the buffer occupancy status, the utility value of the messages, message life time (Tx), and the forwarding policy that supported by SAURP mechanism, such that either the average delivery ratio or delivery delay can be optimized.

The buffer management and scheduling framework with its functionality modules under the heterogeneous nodal mobility scenario is similar to that introduced in section 6.1. The difference is in the way of constructing each model which is based on : i) the network model under consideration, ii) the estimation strategy of the network parameters through NSEM, iii) utility function derivation through UFCM, vi) the forwarding and drop policy, which is incorporated with SAURP mechanism.

6.2.1 Network Model

In this study, a heterogeneous DTN is modeled as a set of N nodes, all moving according to community-based mobility model as descried in chapter 4, where the inter-encounter time between each pair of nodes is an independent and non uniformly distributed. We use the same notation listed in Table 6.1.

The historical inter-encounter time, $\triangle T_{A,B}$, of any two nodes A and B , is calculated according to SAURP mechanism by averaging cumulatively all inter-encounter times until current time-window W^c. Thus, the encounter (or mixing) rate between A and B, is $\beta_{A,B} = \frac{1}{\triangle T_{A,B}}$. According to some social networks studies [66], this value become quite accurate if the nodes have frequent appearance in some locations.

6.2.2 Network State Estimation

Due to the heterogeneity nature of the nodal mobility, it is impractical to gather complete knowledge about the network state (i.e, the values of $m_i(T_i)$, $n_i(T_i)$), instead EHP is employed to estimate these values by disseminating metadata using a fraction of the transfer opportunity. When two nodes encounter each other, they update each other about the messages they have in common, the values of $n_i(T_i)$, $m_i(T_i)$, then the values of $\{\beta_{1,d_i}, \beta_{2,d_i} ... \beta_{n,d_i}\}$, and $\{\beta_{1,d_i}, \beta_{2,d_i} ... \beta_{m,d_i}\}$ are updated accordingly, where β_{n,d_i} and β_{m,d_i} represent the encounter rate between the n^{th} custodian of the n^{th} copy of message i with the destination of message i, and the encounter rate of m^{th} node who has seen the message with the destination of message i, respectively. These parameters are further taken as inputs to calculate the proposed per-message utility function.

The drawback of this method of network state estimation is that it has only imperfect view of the system. The propagated information may be obsolete due to the change in number of copied/dropped messages. Nevertheless, the experiments that we conducted confirmed that this inaccurate information is sufficient to examine the effectiveness of the proposed buffer management policy.

6.2.3 Utility Function Derivation

Maximization of Delivery Ratio

Let us assume that nodes A and B are in contact, and message j in $A's$ buffer is to be forwarded to node B according to SAURP forwarding policy, while the buffer is full at node B and there is a message i with elapsed time T_i in a network that has K messages at the moment at which the decision should be made by node B with respect to dropping a message from all messages in its buffer. To maximize the delivery ratio of all messages, the decision of dropping message i should result in least reduction of delivery probability of message i, while forwarding message j from node A to B should result in increasing the delivery probability of message j. To meet this objective, the decision of dropping message i from the buffer of node B should satisfy following:

$$Umin_i = \left| argmin_i \left[\left(exp(-\sum_{k \in m_i(T_i)} \beta_{k,d_i} T_i) \right) . \right. \right.$$

$$\left. \left(exp(-\sum_{l \in n_i(T_i)} \beta_{l,d_i} R_i) - exp(-\sum_{l \in n_i(T_i) \backslash B} \beta_{l,d_i} R_i) \right) \right] \tag{6.23}$$

and the decision of forwarding message j from node A to node B should satisfy one of two cases; based on whether message j is in spraying phase, or in forwarding phase. If message j is still in spraying phase, the decision of forwarding message j should satisfy following:

$$Umax_j = argmax_j \left[\left(exp(-\sum_{k \in m_j(T_j)} \beta_{k,d_j} T_j) \right) . \right.$$

$$\left(exp(-\sum_{l\in n_j(T_j)}\beta_{l,d_j}R_j) - exp(-\sum_{l\in n_j(T_j)\cup B}\beta_{l,d_j}R_j)\right)\Bigg] \tag{6.24}$$

which represents the margin increase in the delivery probability of message j if node A forward a copy to node B.

If message j is in forwarding phase, the decision of forwarding should satisfy following:

$$Umax_j = argmax_j \left[\left(exp(-\sum_{k\in m_j(R_j)}^{m_j(T_j)}\beta_{k,d_j}T_j)\right).\right.$$

$$\left(exp(-\sum_{l\in n_j(R_j)}\beta_{l,d_j}R_j) - exp(-\sum_{l\in(n_j(R_j)\cup B)\backslash A}\beta_{l,d_j}T_j)\right)\Bigg] \tag{6.25}$$

The relation represents the margin increase in the delivery probability if node A hands over message j to node B.

Proof of (6.23): The probability that a copy of message i will not be delivered by a node is given by the probability that the next meeting time with the destination is greater than its remaining lifetime R_i , assuming that the message i has not yet been delivered. The probability that message i will not be delivered (i.e., none of its copies will be delivered) can be expressed as

$$Pr\{message\,i\,not\,delivered\,|\,not\,delivered\,yet\} =$$

$$P(T_d < T_i + R_i\,|_{T_d>T_i}) = \prod_{l=1}^{n_i(T_i)} exp - (\beta_{l,d_i}R_i) = exp(-(\sum_{l=1}^{n_i(T_i)}\beta_{l,d_j}R_i)) \tag{6.26}$$

By assuming network heterogeneity, the probability that the message is "seen" by each node is not equal. Thus, the probability that message i has been already delivered to the destination is equal to

$Pr\{\, message\, i\, already\, delivered\,\} =$

$$1 - \prod_{k=1}^{m_i(T_i)} exp(\beta_{k,d_i} T_i) = 1 - exp(-(\sum_{k=1}^{m_i(T_i)} \beta_{k,d_j}) T_i) \qquad (6.27)$$

By combining (6.26) and (6.27), the probability that message i is successfully delivered before its Tx expires can be calculated as follows:

$Pr_i = 1 - P\{message\, i\, not\, yet\, delivered\}$

$$*P\{message\, i\, will\, not\, be\, delivered\, within\, R_i\}$$

$Pr_i = 1 - \prod_{k=1}^{m_i(T_i)} exp(-\beta_{k,d_i} T_i) * \prod_{l=1}^{n_i(T_i)} exp(-\beta_{l,d_i} R_i)$

When a node is operating at its maximum buffer capacity, it should drop one or multiple messages so as to achieve the best gain in the increase of the global delivery ratio $Pr = \frac{1}{K(t)} \sum_{i=1}^{K(t)} Pr_i$. To make the optimal decision locally at the node, Pr_i is differentiated with respect to $n_i(T_i)$, and $\partial n_i(T_i)$ is then discretized and replaced by $\triangle n_i(T_i)$.

The best drop policy is one that maximizes $\triangle Pr_i$:

$\triangle Pr_i = \frac{\partial Pr_i}{\partial n_i(T_i)} * \triangle n_i(T_i)$, which is equivalent to
$\triangle Pr_i = \left(exp(-\sum_{k \in m_i(T_i)} \beta_{k,d_i} T_i) \right).$
$\left(exp(-\sum_{l \in n_i(T_i)} \beta_{l,d_i} (R_i) - exp(-\sum_{l \in n_i(T_i) \setminus B} \beta_{l,d_i} R_i) \right)$

Thus, the maximum delivery ratio can be achieved if the message that causes the least decrease in $\triangle Pr_i$ is discarded. On the other hand, when message i is discarded, the number of copies of message i in the network decreases by 1, which results in $\triangle n_i(T_i) = -1$. Thus the optimal buffer dropping policy that can maximize the delivery ratio based on the locally available information at the node is to discard the message with the smallest value of $\left|\frac{\partial Pr_i}{\partial n_i(T_i)}\right|$, which is equivalently to choose a message with a value for Ui_{min} that satisfies (6.23), which represents the marginal decrease in the delivery probability of message i if its copy at node B is dropped.

Proof of (6.24) and (6.25): The proof follows same steps of deriving (6.23) with considering the marginal increase delivery probability of message j at node A if it get copied or forwarded to node B.

Minimization of Average Delivery Delay

To minimize the delivery delay of all messages, the decision of dropping message i should result in least increase of delivery delay of message i, while forwarding message j from node A to B should result in most decrease in the delivery delay of message j (i.e, node B should discard a message such that the expected delivery delay of all messages can be reduced the most). Since the delivery delay of the messages is mainly affected by the nodal inter-encounter time, we assume that all message have infinite or large enough Tx and derive the utility function such that it is affected by number $n_i(T_i)$, $m_i(T_i)$, $\{\beta_{1,d_i}, \beta_{2,d_i} ... \beta_{n,d_i}\}$, and $\{\beta_{1,d_i}, \beta_{2,d_i} ... \beta_{m,d_i}\}$.

To achieve the minimum average delivery delay, node B should drop the message that satisfies the following:

$$Umin_i =$$

$$argmin_i \left| \left[exp(- \sum_{k \in m_i(T_i)} \beta_{k,d_i} T_i) \left(\frac{1}{\sum_{l \in n_i(T_i)} \beta_{l,d_i}} - \frac{1}{\sum_{l \in n_i(T_i) \setminus B} \beta_{l,d_i}} \right) \right] \right| \tag{6.28}$$

and the decision of forwarding message j from node A to node B should satisfy one of two cases; based on whether message j is in spraying phase, or in forwarding phase. If message j is still in spraying phase, the decision of forwarding message j should satisfy following:

$$Umax_j =$$

$$argmax_j \left[exp(-\sum_{k \in m_j(T_j)} \beta_{k,d_j} T_j) \left(\frac{1}{\sum_{l \in n_j(T_j)} \beta_{l,d_j}} - \frac{1}{\sum_{l \in n_j(T_j) \cup B} \beta_{l,d_j}} \right) \right] \tag{6.29}$$

which represents the margin decrease in the delivery delay of message j if node A forward a copy to node B.

If message j is in forwarding phase, the decision of forwarding should satisfy following:

$$Umax_j =$$

$$argmax_j \left[exp(-\sum_{k \in m_j(R_j)}^{m_j(T_j)} \beta_{k,d_j} T_j) \left(\frac{1}{\sum_{l \in n_j(T_j)} \beta_{l,d_j}} - \frac{1}{\sum_{l \in (n_j(R_j) \setminus A) \cup B} \beta_{l,d_j}} \right) \right] \tag{6.30}$$

The relation represents the margin decrease in the delivery delay if node A hands over message j to node B.

Proof of (6.28): Let random variable T_d represents the delivery delay of message i. Then, the expected delay in delivering a message that still has copies existing in the network can be expressed

$D_i = P\{message\, i\, not\, deliverd\, yet\} * E[T_d \mid T_d > T_i]$

$$D_i = exp(-\sum_{k \in m_i(T_i)} \beta_{k,d_i} T_i) * E[T_d \mid T_d > T_i] \tag{6.31}$$

where

$$E[T_d \mid T_d > T_i] = \left[T_i + \frac{1}{\sum_{l \in n_i(T_i)} \beta_{l,d_i}} \right] \tag{6.32}$$

When a node buffer is full, the node should make a drop decision that leads to the least increase on D_i. To find the local optimal decision, D_i is differentiated with respect to $n_i(T_i)$, and ∂D_i is then discritized and replaced by $\triangle D_i$:

$\triangle D_i = \frac{\partial D_i}{\partial n_i(T_i)} * \triangle n_i(T_i)$, which equivalent to
$\triangle D_i = exp(-\sum_{k \in m_i(T_i)} \beta_{k,d_i} T_i) \left[\frac{1}{\sum_{l \in n_i(T_i)} \beta_{l,d_i}} - \frac{1}{\sum_{l \in n_i(T_i) \setminus B} \beta_{l,d_i}} \right] \triangle n_i(T_i)$

To reduce the delivery delay of all messages existing in the network, the best decision is to discard the message that maximizes the total delivery delay, $D = \sum_{i=1}^{K(t)} D_i$, among all the messages. Therefore, the optimal buffer-dropping policy at node B that leads to minimization of the delivery delay is thus to discard the message that has the min value of $\mid \triangle D_i \mid$ (or$-\triangle D_i$), which is equivalently to choose a message with a value for Ui_{min} that satisfies (6.28), which represents the marginal increase in the delivery delay of message i if its copy at node B is dropped, while the optimal buffer-forwarding policy at node A that leads to minimization of the delivery delay is thus to forward a copy of message j (or message j itself) to node B that leads to the maximum increase of $\triangle D_i$, which is equivalently to choose a message with a value for Uj_{max} .

Proof of (6.29) and (6.30): The proof follows same steps of deriving (6.28) with considering the marginal decrease of delivery delay of message j at node A if it get copied or forwarded to node B.

Algorithm 6.1 SAURP_based forwarding and dropping policy

On contact between node A and B
Exchange summary vectors
If (buffer at node B is full)
    for every message j at the buffer of custodian node A *do*
        if (B is not source node of i) *then*
            if (remaining tokens of message $j \geq$ remaining tokens of i) &&
            $(\triangle T_{B,d_i} \succ min\{\triangle T_{1,d_i}, \triangle T_{2,d_i} ...., \triangle T_{n-1,d_i}\}) then$
                if destination node d_j in transmission range of B *then*
                    B drops message i
                    A forwards a copy of message j to B
                end if
                else if $(Umax_j - Umin_i > 0$ *) then*
                    B drops message i
                    A forwards message j to B
                end else if
            end if
        end if
    end for
end if
else (apply SAURP)
end

6.2.4 SAURP_based Forwarding and Dropping Policy (SFDP)

With the per-message utility, the node firstly sorts the buffer messages accordingly from the highest to the lowest. The messages with lower utility values have higher priorities to be dropped when the node's buffer is full, while the messages with higher utility values have higher priorities to be forwarded to the encountered node. Algorithm 6.1. illustrates the forwarding and dropping actions which largely based on the fact that; if the utility $Umax_j$ of message j (the message with the highest utility value) buffered in A is higher than $Umin_i$ of message i (the message with the lowest utility value) at node B, then message i is dropped and replaced by message j or copy of it, if the buffer of B is full during the contact of the two nodes.

6.2.5 Evaluation Scenario

To examine the efficiency of the proposed SAURP_based Forwarding and Dropping Policy (SFDP), experiments were conducted, and the results presented in this subsection. As we justified earlier in section 6.1, the EHP strategy gives sub-optimal solution, nonetheless it is the most practical strategy to be implemented for estimating the values of $m_i(T_i)$, and $n_i(T_i)$ in such heterogeneous decentralized environment. We call SFDP under EHP strategy as SFDP_E. To better understand the performance of the proposed strategies and their gain over SAURP, we also implemented the Global Knowledge-based Management (GKM) strategy (introduced in 6.2.1) for estimation the values of $m_i(T_i)$, and $n_i(T_i)$. We call SFDP under GKM strategy as SFDP_G. The result of SFDP under GKM is taken as a benchmark for the proposed schemes, since such an assumption of global knowledge is not practical [80].

In addition to the above prediction strategy, we compared the proposed buffer management policies with two well-known forwarding and scheduling schemes listed as follows:

- Delegation forwarding scheme employes dropping policy based on drop message with highest number of forwards (DF_N) by Erramilli et al. in [80].
- RAPID scheme employs dropping policy based on drop message that is most likely to miss the deadline [69].

We assume a message issued at a node (termed sourced messages) has the highest priority at the node. If all buffered messages are sourced ones and the newly arrived message is also a source message at the node, then the oldest one is dropped. This idea was examined in [33] and has been proved with improved delivery ratio.

Table 6.3: Simulation parameters

Mobility pattern	CBMM	Infocom06
Simulation duration (seconds)	30000	270000
Simulation area	700×700	–
No. of Nodes	110	98
TTL (sec)	9000	75000

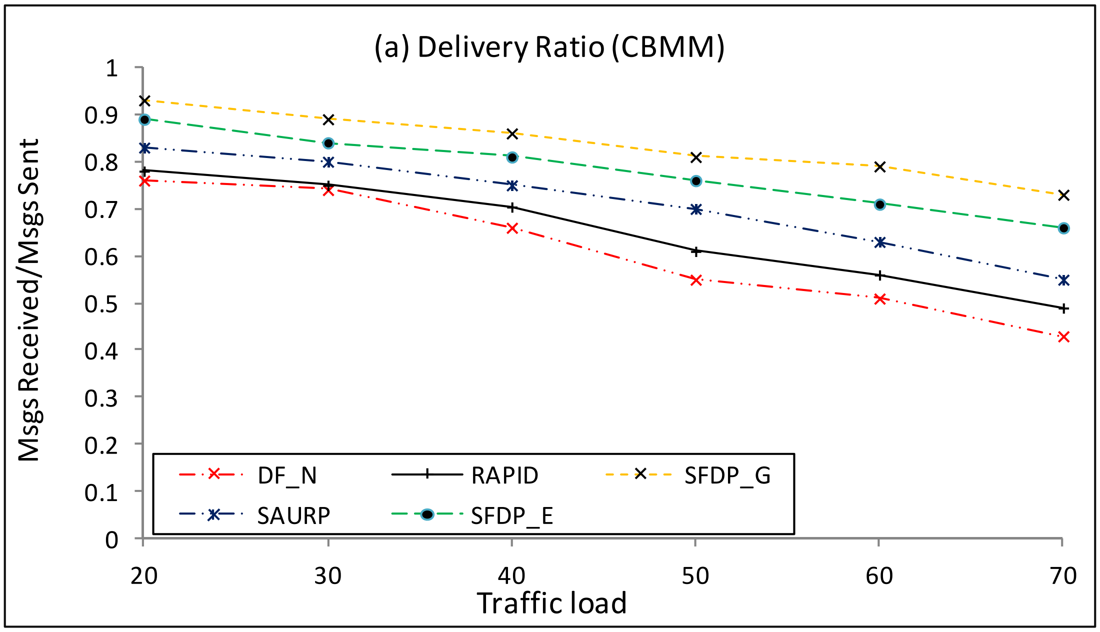
Experimental Setup

The simulations are based on two mobility scenarios; a synthetic one based on the same mobility scenario of chapter 4, and a real-world encounter traces collected as part of the Infocom 2006 experiment. These data sets comprise of contact traces between short-range Bluetooth enabled devices carried by individuals in Infocom 2006 conference environment. More details about the devices and the data sets, including synchronization issues can be found in [104]. In order to observe the performance impact and how the proposed policies react under congested environment, we set the bandwidth, buffer capacity, and the distribution of the contact time such that congested environment is formed. The transmission range is set to 20 for CBMM scenario, the channel bandwidth is set relatively quite small (i.e., one message transfer per unit of time), and the buffer size is set to 10, under different levels of traffic demand. Message delivery ratio and the delivery delay are taken as two performance measures. Each data is the average of the results from 30 runs.

Introduced Policy for Maximizing Delivery Ratio

In all scenarios, the traffic load varies from 20 to 70 messages generated per node. The plots of the delivery ratio obtained for all policies is shown in Figure 6.9.

It can be seen in Figure 6.9(a) that SFDP_G gives the best performance, which



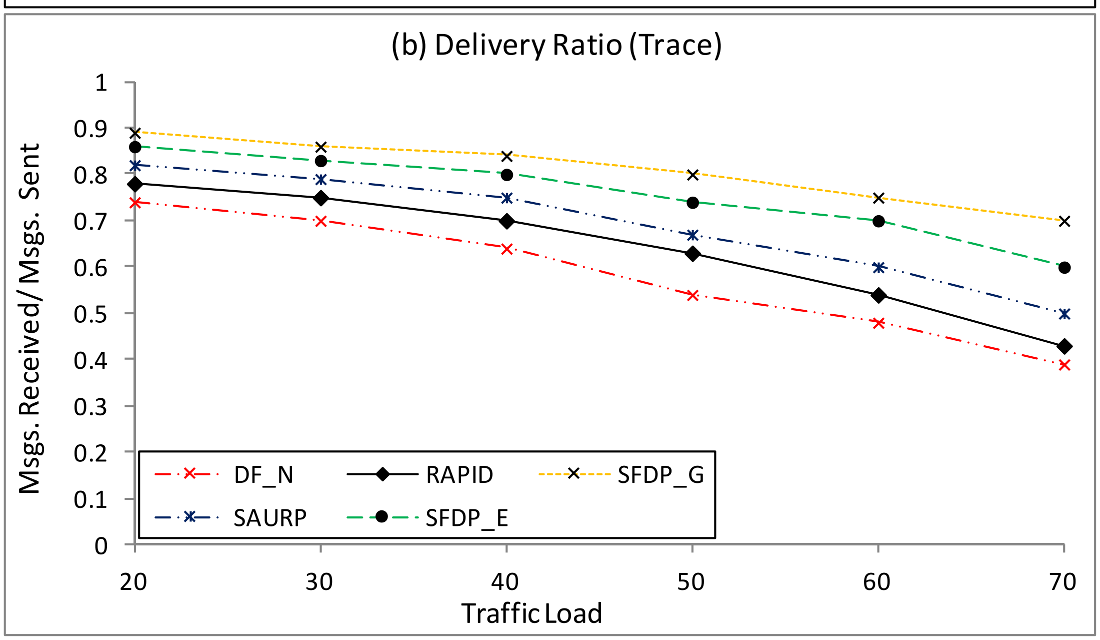


Figure 6.9: The effect of traffic load on the delivery ratio

meets our expectation. The SFDP_E policy provides the next best result and is competitive with the SFDP_G in the case of low traffic. As the traffic increases, the performance of all policies degrades, while the SFDP_E outperforms SAURP, RAPID, and DF_H. It can achieve a delivery ratio 28% higher than that achieved by SAURP, 42% than that of DF_N, 35% than that by RAPID, and 16% worse than SFDP_G. Figure 6.9(b) shows the results of delivery ratio under the real trace. As can be seen, the SFDP_E policy provides the next best result right after SFDP_G. SFDP_E still outperforms SAURP, RAPID, and DF_N. At high traffic loads, It can achieve delivery ratio higher than SAURP by 26%, RAPID by 33%, DF_N by 40%, and worse than SFDP_G by 12% .

Introduced Policy for Minimizing Delivery Delay

Figure 6.10 shows the results for the delivery delays. As expected, the SFDP_G gives the best performance under all traffic loads for both scenarios under consideration, while the SFDP_E is the second best and is competitive with the SFDP_G in the case of low traffic. As the traffic increases, the demand on the wireless channel and buffers increases, causing a long queuing delays and substantial message loss that negatively affect the performance of all the examined policies. Figure 6.10(a) shows the results under CBMM scenario. We have observed that the SFDP_H outperforms the SAURP, DF_N, and RAPID. SFDP_E is better than SAURP by 21%, RAPID by 35%, DF_N by 44% and a longer delay of only 23% of that achieved by SFDP_G. Under the real trace scenario as shown in Figure 6.10(b), SFDP_E achieved delivery delay better than SAURP by 27%, RAPID by 43%, DF_N by 56%, and a longer delay of only 13% of that achieved by SFDP_G.

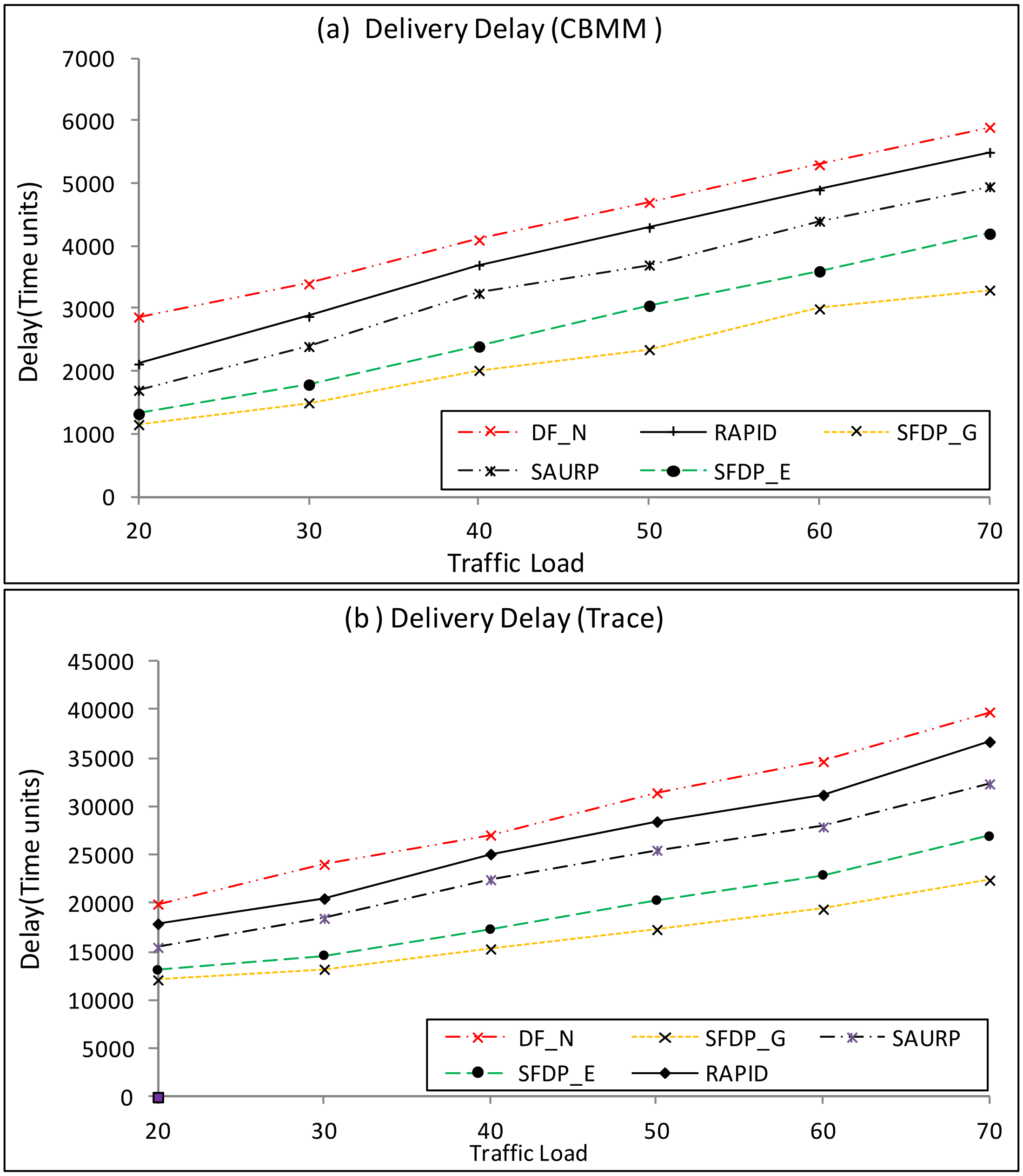


Figure 6.10: The effect of traffic load on the delivery delay

6.3 Summary

In this chapter, we investigated a novel buffer management framework for two families of routing; flooding-based, and utility-based routing. Epidemic and source forwarding routing are considered for homogeneous DTNs, while SAURP is considered for heterogeneous DTNs, aiming to optimize either the message delivery ratio or message delivery delay. The introduced framework incorporates a suit of novel mechanisms for network state estimation and utility derivation, such that a node can obtain the priority for dropping each message in case of buffer full. The simulation results show that the proposed buffer management framework can significantly improve the routing performance in terms of the performance metrics of interest under limited network information.

Chapter 7

Conclusion and Future Research

7.1 Conclusion

The objective of this research is to achieve end-to-end data delivery over intermittently connected mobile networks. Regular MANETs protocols fail to provide successful communications due to user's frequent disconnections and long disconnection periods. This research has presented our studies and has provided a suit of solutions to problems of routing in DTNs. The introduced self-adaptive routing protocol (SARP) for opportunistic DTNs can effectively reduce data delivery delay and bandwidth consumption. Based on this work, the research has been expanded to cover the routing problem for highly congested DTNs. Two contention aware routing techniques were proposed, self adaptive utility-based routing Protocol (SAURP), and adaptive reinforcement-based routing protocol (ARBRP). The work is further expanded to investigate a new buffer management and scheduling framework for flooding and encounter based routing in DTNs, such that the forwarding/dropping decision can be made at a node during each contact for either optimal message delivery ratio or message delivery delay. The accomplishments in this thesis are summarized as follows:

- Introducing a novel routing protocol called self adaptive routing protocol (SARP) [63]. The introduced solution has the goal of investigating the effect of deploying a self-organized framework for routing messages in sparsely connected mobile networks. The protocol is characterized by employing an efficient updating strategy for the stochastic information at each node. It can achieve minimum delivery delay, high delivery ratio, and low transmissions. SARP is designed to alleviate the shortcomings of existing utility-based protocols that designed for networks that have sufficient resources, and/or lightly loaded.

- Introducing two novel contention aware routing techniques, called self adaptive utility-based routing protocol (SAURP) [85, 103], and adaptive reinforcement-based routing protocol (ARBRP) [64] for DTNs. Each routing scheme employs the exploitation of the network state information and the nodal mobility in a different way. SAURP uses a utility function in a form of inter-contact time as the main factor on its forwarding decision, while ARBRP employs a utility function in a form of contact time duration as the main factor on its forwarding decisions. The main feature of the introduced protocols is the strong capability in adaptation to the fluctuation of network status, traffic patterns/characteristics, and user behaviors, so as to reduce the number of transmissions, message delivery time, and increase delivery ratio. This is achieved by jointly considering node mobility statistics, congestion, and buffer occupancy, which are subsequently fused in a novel quality-metric (utility) function. In specific, the link availability and buffer occupancy statistics are obtained by sampling the channels and buffer space during each contact with another node. The developed quality-metric function targets to facilitating decision making for each active data message, resulting in optimized network performance.

- Introducing novel message scheduling framework to enhance the performance of flooding and controlled flooding forwarding routing, in which additional buffer space and bandwidth overhead are needed in order to increase message delivery ratio and/or reduce message delivery delay. In specific we introduced new message scheduling framework for epidemic and source forwarding routing in delay tolerant networks [82], such that the forwarding/dropping decision can be made at a node during each contact for either optimal message delivery ratio or message delivery delay.

- Introducing a new message scheduling framework for utility-based forwarding routing in delay tolerant networks. In specific, we develop buffer management policy based on the mechanism of the SAURP. The decision of forwarding or dropping the buffered messages is made based on the buffer occupancy status, the utility value of the messages and the forwarding policy that supported by SAURP mechanism, such that either the average delivery ratio or delivery delay can be optimized.

7.2 Future Research Plan

To achieve close to optimal performance in an intermittently connected environment, it is necessary to use more than one copy. The understanding acquired from the detailed examination of single-copy schemes has been employed to develop and design efficient multi-copy schemes that can achieve the desired performance in a large range of scenarios. However, more work needs to be accomplished. Future research plan will focus on the following issues:

- The work can be expanded to include the cases of multiple-copy routing, and the cases when the sending of multiple copies of a message throughout a network becomes a requirement for meeting delivery requirements. Such protocol should overcome limitations that still exist in multi-copy routing schemes, such as node failure or fault tolerance, and meeting the delivery deadline required for some applications. To meet the delivery deadline for some messages, the copies of a message should be distributed among the potential relays in such a way that predefined percentage of all messages meet the given delivery deadline with a minimum number of copies spread throughout the network.

- Acquiring the global knowledge about network state to enhance buffer management policies is still an open research issue. All of the proposed buffer management schemes for heterogeneous DTN provide sub-optimal solutions because they consider only partial information about the nature of the DTN. Thus, developing strategy for acquiring global knowledge through already deployed networks such as GSM, or WIMAX is worthy to investigate. This case all control traffic may goes over a low-bandwidth, long range radio, while actual date transfer goes during the encounters between nodes.

- Several utility functions presentations have been introduced in order to deal with routing issues in encounter-based routing in DTNs. These utility functions have different information presentations and have significant impact on the behavior of the protocols. To examine the impact of these utility functions on the behavior of the protocols, the levels of search diversity and convergence that are affected by the utility functions should be assessed. Future work may include the proposal of an assessment technique for describing the level of search diversity and convergence. To the best of our knowledge, no technique has been applied in order to assess the diversity and convergence of routing protocols in encounter-based DTN routing.

- Load Balancing is very important issue. The typical state-of-the-art routing algorithms for delay tolerant networks are based on best next hop in order to achieve throughput and efficiency. Based on studies on the routing messages, specially that based on social network structure, the traffic of messages mostly is directed of through a small subset of good users. This unfair load distribution is not sustainable as it can quickly deplete constraint resources in heavily utilized mobile devices (e.g. storage, battery, budget, etc.). Moreover, because a small number of users carry a significant amount of the traffic, the system is not robust to random failures and attacks. Thus proposing new techniques that take this issues is the consideration is one of research directions that worth to investigate.

- Developing new strategy to disseminate backward acknowledgments to delete messages from custodian nodes when one of their copies reach its destination.

Appendix A

Routing

proof of (5.10):

first we compute the convolution needed in the proof.

$e^{-ax} \otimes e^{-bx} = \int_0^x e^{-a(x-u)} e^{-bu} du = e^{-ax} \frac{e^{(a-b)x}-1}{a-b} = \frac{e^{-bx}-e^{-ax}}{a-b}$

for two hop (n=2)

$P_{1+2} = P_1 \otimes P_2 = \beta_1 \beta_2 \int_0^x e^{-(\beta_1-\beta_2)t} e^{-\beta_2 x} dt$

$= \frac{\beta_1 \beta 2}{\beta_2-\beta_1} \left(e^{-(\beta_1 x - \beta_2 x)}\right) = \beta_1 \beta 2 \left[\frac{e^{-\beta_1 x}}{\beta_2-\beta_1} + \frac{e^{-\beta_2 x}}{\beta_1-\beta_2}\right] = -C_1^{(2)} P_1(x) - C_2^{(2)} P_2(x)$

For $k \geq 3$, inductively we can get

$P_{k-1} = \sum_{i=1}^{k_l-1} C_i^{k_l-1} . P_i(x)$

$P_k = P_{1+2+\ldots k-1} \otimes P_k = \left[\prod_{i=1}^{k-1} \beta_i\right] \sum_{j=1}^{k_l-1} \frac{e^{\beta_j x}}{\prod_{i=1}^{k-1}(\beta_k-\beta_j)} \otimes P_k$

$= \sum_{i=1}^{k_l-1} C_i^{k_l-1} \left(\frac{\beta_k}{\beta_k-\beta_i} P_i(x) + \frac{\beta_i}{\beta_i-\beta_k} P_k(x)\right)$

if we consider $C_i^{k_l} = C_i^{k_l-1} . \frac{\beta_i}{\beta_i-\beta_k}$, we get

$P_k = \sum_{i=1}^{k_l-1} C_i^{k_l} . P_i(x) + \sum_{i=1}^{k_l-1} C_i^{k_l-1} . \frac{\beta_i}{\beta_i-\beta_k} P_k(x)$

for second term , we have $\sum_{i=1}^{k_l-1} C_i^{k_l-1} \frac{\beta_i}{\beta_i-\beta_k} = \sum_{i=1}^{k_l-1} \frac{\beta_i}{\beta_i-\beta_k} . \left(\prod_{j=1,\, j\neq i}^{k-1} \frac{\beta_j}{\beta_j-\beta_i}\right)$

$= \prod_{j=1}^{k-1} \beta_j . \sum_{i=1}^{k_l-1} \prod_{j=1,\, j\neq i}^{k-1} \frac{1}{\beta_j-\beta_i} = \prod_{j=1}^{k-1} \frac{\beta_j}{\beta_j-\beta_k} = C_k^{k_l}$

Therefore, we have

$f_k(x) = P_k = \sum_{i=1}^{k_l-1} C_i^{k_l} P_i(x) + C_k^{k_l} P_k(x) = \sum_{i=1}^{k_l} C_i^{k_l} P_i(x)$

CDF:

$F_k(x) = P(T_d < T) = \int_0^T f_k(x)dx = \sum_{i=1}^{k_l} C_i^{k_l} \int_0^T P_i(x)dx = \sum_{i=1}^{k_l} C_i^{k_l}.(1 - e^{-\beta_i T})$

CCDF:

$= P(Td > T) = \int_T^\infty f_k(x)dx = \sum_{i=1}^{k_l} C_i^{k_l}.e^{-\beta_i T}$

The expected delay when T goes to infinity is

$E_l[x] = \int_0^\infty P(Td > T) = \sum_{i=1}^{k_l} C_i^{k_l}. \int_0^\infty e^{-\beta_i t}dt = \sum_{i=1}^{k_l} C_i^{k_l}.\frac{1}{\beta_i}$

The total CCDF for l paths from source to destination for one hop relay is calculated as below:

The CDF is calculated as:

$P_l(T_d < T) = 1 - \prod_{j=1}^{l}(1 - p_j(T_d < T) =$

$P_l(T_d < T) = 1 - e^{(-\sum_{j=1}^{l} \beta_j T)}$, where β is the encounter rate between custodian node j and the destination

Therefore, the CCDF $P(T_d > t_i) = e^{(-\sum_{j=1}^{l} \beta_j t_i)}$

If each path has k_l number of hops, the CCDF formula for one path would be:

$= P(T_d > T) = \sum_{i=1}^{k_l} C_i^{k_l}.e^{-\beta_i T}$

CCDF formula in case of L paths would be

$= P_L(T_d > T) = \prod_{p=1}^{l} \sum_{i=1}^{k_p} C_i^{k_p}.e^{-\beta_{pi} T}$

Therefore, the total expected delay for l paths is:

$Em[T_d] = \int_0^\infty P_L(T_d > t) = \int_0^\infty (\prod_{p=1}^{l} \sum_{i=1}^{k_p} C_i^{k_p}.e^{-\beta_{pi} t})dt$

Appendix B

Buffer Management and Scheduling

Proof of (6.10): Given $m_i(T_i)$, $n_i(T_i)$,and $P(T_i) = \frac{m_i(T_i)}{N-1}$, as initial values at T_i , the delivery probability in the interval $t : T_i < t < T_i + R_i$, $P(T_d < T_i + R_i \mid_{T_d > T_i})$, can be constructed using (6.3) as follows:

$P_i'(t) = \frac{dP}{dt} = \beta n_i(t)(1 - P_i(t))$
$\frac{dP}{1-p} = \beta n_i(t)dt$
$\frac{dP}{1-p} = \beta \frac{Nn_i(0)}{n_i(0)+(N-n_i(0))e^{-P_{fi}\beta Nt}} dt$

Integrate both sides for the interval R_i, we get

$$P(T_d < T_i + R_i \mid_{T_d > T_i}) = 1 - \left[\left(1 - \frac{m_i(T_i)}{N-1} \right) \right.$$

$$\left. * \left(\frac{N}{N - n_i(T_i) + n_i(T_i)e^{\beta P_{fi} N(Ri)}} \right)^{\frac{n_i(T_i)}{P_f}} \right]$$

Proof of (6.13): Delivery probability within R_i and the initial state is at T_i

Calculating T_L value:

Given $n_i(T_i)$ we can expect the time, T_L, at which L message copies in the network are spread as following:

$$n(T_L{-}T_i) = N - (N - n(T_i))e^{-\beta P_f(T_L - T_i)}$$
$$\frac{N-L}{N-n(T_i)} = e^{-\beta P_f(T_L - T_i)} \rightarrow -\beta P_f(T_L{-}T_i) = Ln\frac{N-L}{N-n(T_i)}$$
$$T_L - T_i = \frac{1}{\beta P_f}Ln(\frac{N-n(T_i)}{N-L})$$
$$T_L = T_i + \frac{1}{\beta P_f}Ln(\frac{N-n(T_i)}{N-L})$$

- Delivery within R_i and $P(T_i) = \frac{m_i(T_i)}{N-1}$,

Two cases are identified: (1) $T_i < T_{L_i}$, (2) $T_{Li} \leq T_i \leq R_i$.

Case (1): $T_i < T_{Li}$, which has two periods $(T_i, T_L)$and $(T_L, Tx - T_{Li})$

1-Period (T_i, T_{Li}) :

We have: $\frac{dP(t)}{dt} = P'(t) = \beta n(t)(1 - P(t))$ with initial conditions $P(T_i)$ and $n(T_i)$.

$\frac{dP}{1-P} = \beta\left[N - (N - n(T_i))e^{-\beta P_f t}\right]$. Integrating both sides for the interval $T_{Li} - T_i$, we get

$$P(T_d \leq T_{Li} \mid T_d \geq T_i) = 1 - \Big[(1 - P(T_i))\, e^{\beta N(T_{Li} - T_i)}$$

$$* e^{\frac{1}{P_f}(N-n(T_i))e^{-\beta P_f(T_{Li}-T_i)} - \frac{1}{P_f}(N-n(T_i))}\Big],\ \ T_i < T_{Li}$$

2-For period (T_L, Tx):

$\frac{dP(t)}{dt} = P'(t) = \beta L(1 - P(t))$

$\frac{dP}{1-P} = \beta L dt$. Integrating both sides for the interval $Tx - T_{Li}$, we get

$$P(Tx - T_{Li}) = 1 - \left[(1 - \tfrac{m_i(T_i)}{N-1})e^{\beta N(T_{Li}-T_i)} * e^{\frac{1}{P_f}(N-n(T_i))e^{-\beta P_f(T_{Li}-T_i)} - \frac{1}{P_f}(N-n(T_i))}\right] e^{-\beta L(Tx-T_{Li})}$$

Therefore the total delivery probability at $T_i < R_i$ is given by

$$P(T_{Li} < T_d \leq Tx - T_{Li}) = 1 - \left[(1 - \tfrac{m_i(T_i)}{N-1})e^{\beta N(T_{Li}-T_i)} * e^{\beta\frac{1}{P_f}(N-n(T_i))(e^{-\beta P_f(T_{Li}-T_i)}-1)}\right] e^{-\beta L(Tx-T_{Li})}$$

Case (2): $T_i \geq T_{Li}$

The initial condition $P(T_i) = \frac{m_i(T_i)}{N-1}$.

$\frac{dP(t)}{dt} = P'(t) = \beta P_f L(1 - P(t))$ (multiply by P_f to consider the situation the message with L copies is most likely get dropped)

$\frac{dP}{1-P} = \beta L dt$. Integrating both sides for the interval $Tx - T_i = R_i$:

$$\int_{P(T_i)}^{P(Tx)} \frac{1}{1-P} dp = \beta P_f L \int_0^{R_i} dt$$

The final expression is:

$$P(T_i < T_d < T_i + R_i) = 1 - \left[1 - \tfrac{m_i(T_i)}{N-1}\right] e^{-\beta P_f L R_i}$$

Proof of (6.18):

$E[T_d \mid T_d > T_i] = T_i + \int_0^{\propto} (1 - (P(t))dt$
$E[T_d \mid T_d > T_i] = T_i + \int_0^{\propto} \left(\frac{N}{N - n(T_i) + e^{\beta P_f N T i}} \right)^{\frac{1}{p_f}} dt$

$$E[T_d] = \frac{-1}{\beta N} \left((N - n(T_i))e^{-\beta P_f N t} \right)^{\frac{1}{P}}$$

$$* \left(\frac{N}{N - n(T_i) + e^{\beta P_f N t}} \right) .F \mid_0^{\infty}$$

According to the saddle point approximation [33], the final formula is obtained as

$E[T_d] = \frac{lnN}{\beta P_f (N - n(T_i))}$

The expected delivery delay at any elapsed time instance

$$E[T_d \mid T_d > T_i] = T_i + \frac{Ln(N)}{\beta P_f (N - n(T_i))}$$

Proof of (6.22):

$D_i = P\{message\, i\, not\, deliverd\, yet\} * \frac{1}{P_f} E[T_d \mid T_d > T_i].$
$E[T_d \mid T_d > T_i] = \left[T_i + \int_0^{\infty} t f(t) dt \right]$

since the mobility and nodes are exponentially distributed, the respected delay of a message can be calculated as

$\left[T_i + \int_0^{\infty} t \beta e^{-\beta t} dt \right] = T_i + \frac{1}{\beta}.$

In case of a message is carried by L nodes

$E[T_d] = T_i + \frac{1}{L\beta}$

considers the case when the network has $n_i(T_i) < L$, with message remaining life time R_i. With probability $\frac{m_i(T_i)}{N-1}$ the next encountered node is the destination node, and with probability $1 - \frac{m_i(T_i)}{N-1}$ the next encountered node is not the destination and will get a copy from the source node. The expected time that source node is waiting to encounter any other node (stays at state s), ED is given by

$$ED = \frac{1}{(N-1)\beta} \text{ , } ED(n(T_i)+1) = \frac{1}{(n(T_i)+1)\beta}.$$

Based on the analysis above, the expected delay of a message of source forwarding scheme can be given as follows:

$$T_i + \frac{1}{P_f}\left[\frac{m_i(T_i)}{N-1}(ED) + \left(1 - \frac{m_i(T_i)}{N-1}\right) ED\left(n_i(T_i)+1\right)\right],\ T_i < T_L$$

Since we are interested in the copy of messages that are not yet been delivered, the above expression is written as

$$E[T_d \mid T_d > T_i] = T_i + \frac{1}{P_f}\left[\left(1 - \frac{m_i(T_i)}{N-1}\right) ED\left(n_i(T_i)+1\right)\right], T_i < T_L$$
$$E[T_d \mid T_d > T_i] = T_i + \frac{1}{P_f}\left[\left(1 - \frac{m_i(T_i)}{N-1}\right) . \frac{1}{(n(T_i)+1)\beta}\right], T_i < T_L$$

for $n(T_i) = L$,

$E[T_d \mid T_d > T_i] = T_i + \frac{1}{P_f L\beta}$

The final formula of (6.22) is derived by combining the the above equations with probability of a message not yet delivered for either cases $T_i < T_L$ or $T_i \geq T_L$.

$$
D_i = \begin{cases} \left(1 - \frac{m_i(T_i)}{N-1}\right) * \left[T_i + \frac{1}{P_f}\left(1 - \frac{m_i(T_i)}{N-1}\right) * \right. \\ \left. \frac{1}{(n(T_i)+1)\beta}\right], \ T_i < T_L \\ \\ \left(1 - \frac{m_i(T_i)}{N-1}\right)\left[T_i + \frac{1}{P_f L \beta}\right], \quad T \geq T_L \end{cases}
$$

References

[1] Delay Tolerant Networking. http://www.dtnrg.org. 2, 7, 8, 14

[2] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, and K. Scott. Delay-tolerant networking: an approach to interplanetary Internet. IEEE Communications Magazine, 2003. 2

[3] Disruption tolerant networking. http://www.darpa.mil/ato/solicit/DTN/. 2

[4] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Pocket switchednetworks: Real-world mobility and its consequences for opportunistic forwarding. Technical Report UCAM-CL-TR-617, University of Cambridge, 2005. 2

[5] R. Groenevelt, P. Nain, and G. Koole. 2005. Message delay in MANET. In Proceeding of the ACM SIGMETRICS, 2005. 16, 38, 42, 76, 110, 115, 117

[6] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein. Energy- efficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet. In Proceedings of ASPLOS, 2002. 2, 35, 129

[7] A. Seth, D. Kroeker, M. Zaharia, S. Guo, and S. Keshav. Low-cost communication for rural internet kiosks using mechanical backhaul. In Proceedings MobiCom, 2006. 2

[8] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In Mobile Computing and Networking, pages 85–97, 1998. 2

[9] Z. Zhang. Routing in Intermittently Connected Mobile Ad Hoc Net-works and Delay Tolerant Networks: Overview and Challenges. IEEE Communications Surveys and Tutorials, 8(1), 2006. 7, 8, 38

[10] K. Fall. A delay–tolerant network architecture for challenged internets. In Proceedings of ACM SIGCOMM, 2003. 7, 8

[11] E. P. Jones and P. A. Ward. Routing strategies for delay-tolerant networks. Submitted to Computer Communication Review, 2006. 7, 103

[12] S. Jain, K. Fall, and R. Patra. Routing in a delay tolerant network. In Proceedings of ACM SIGCOMM, 2004. 9, 12

[13] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data deliveryin sparse mobile ad hoc networks. In Proceedings of MobiHoc, 2004. 10

[14] Q. Li and D. Rus. Communication in disconnected ad hoc networks using message relay. Journal of Parallel and Distributed Computing, 63(1), 2003. 10

[15] W. Zhao, M. Ammar, and E. Zegura.Controlling the mobility of multiple data transport ferries in a delay-tolerant network. In Proceedings of IEEE INFOCOM 2005. 2

[16] D. Marasigan and P. Rommel. MV routing and capacity building in disruption tolerant networks. In Proceedings of IEEE INFOCOM, 2005. 11

[17] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Utility-based Message Replication for Intermittently Connected Heterogeneous Wireless Networks. In Proceedings of IEEE WoWMoM workshop on Autonomic and Opportunistic Communications (AOC), 2007. 11

[18] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networking. In Proceedings of IEEE INFOCOM, 2006. 12

[19] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft and C. Diot. Pocket switched networks and human mobility in conference environments. In Proceedings of the ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN'05), August 2005. 12

[20] J. N. Al-Karaki and A. E. Kamal. Routing techniques in wireless sensor networks: a survey. IEEE Wireless Communications, vol. 11, no. 6, pp. 6–28, 2004. 12

[21] H. Jun, M. H. Ammar, and E. W. Zegura. Power management in delay tolerant networks: A framework and knowledge-based mechanisms. In Proceedings of IEEE SECON, 2005. 13

[22] R. C. Shah, S. Roy, S. Jain, and W. Brunette. Data mules: Modeling and analysis of three-tier architecture for sparse sensor networks. Elsevier Ad Hoc Networks Journal, 1:215–233, Sept. 2003. 10

[23] G. Sharma and R. Mazumdar. On achievable delay/capacity trade-offs in mobile ad hoc networks. In Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (Wi'Opt), 2004. 17

[24] B. An and S. Papavassi. An Entropy-Based Model for Supporting and Evaluating Route Stability in Mobile Ad hoc Wireless Networks. IEEE Communications Letters, VOL.6, NO. 8, AUGUST 2002. 18

[25] Y. Wang, S. Jain, M. Martonosi, and K. Fall. Erasure coding based routing for opportunistic networks. In Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005. 35

[26] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss. Delay-tolerant networking architecture, RFC4838, RFC Editor, April 2007. 2

[27] T. Small and Z. J. Haas. Resource and performance tradeoffs in delay-tolerant wireless networks. In Proceedings of the ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN'05), 2005. [Online]. Available: http://dx.doi.org/10.1145/1080139.1080144 16, 17

[28] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In Proceedings of the ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN'05), 2005. 16, 17, 20, 22, 23, 25, 55

[29] A. Vahdat and D. Becker. Epidemic routing for partially-connected ad hoc networks. Duke University, Tech. Rep. CS-2000-06, July 2000. [Online]. Available: http://issg.cs.duke.edu/epidemic/epidemic.pdf 16, 17, 41, 42, 55, 95, 110, 117

[30] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu. The broadcast storm problem in a mobile ad hoc network. Wireless Networks, 8(2/3), 2002. 17, 18, 20, 35

[31] P. Hui, J. Crowcroft. How Small Labels create Big Improvements. Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops. (PerCom), 2007. 17

[32] Q. Li and D. Rus. Communication in disconnected ad hoc networks using message relay. Journal of Parallel and Distributed Computing, 63(1), 2003. 17, 23

[33] X. Zhang, G. Neglia, J. Kurose, and D. Towsley. Performance modeling of epidemic routing. Technical Report CMPSCI 05-44, UMass, 2005. 18, 32, 111, 112, 113, 129, 148, 164

[34] J. Widmer and J.-Y. L. Boudec. Network coding for efficient communication in extreme networks. In Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN), 2005.

[35] K. A. Harras and K. C. Almeroth. Transport layer issues in delay tolerant mobile networks. In Proceedings of IFIP-TC6 Networking, May 2006. 17

[36] M. Mauve, A. Widmer, and H. Hartenstein. A survey on position-based routing in mobile ad hoc networks. IEEE Network, vol. 15, no. 6, pp. 30–39, 2001. [Online]. Available: http://dx.doi.org/10.1109/65.967595 18

[37] T. S. E. Ng and H. Zhang. Predicting internet network distance with coordinates-based approaches. In Proceedings of IEEE INFOCOM, 2002. 19

[38] B. Karp and H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In Proceedings of ACM MobiCom, 2000. 19

[39] J. Lebrun, C.-N. Chuah, D. Ghosal, and M. Zhang. Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks. In Proceedings of IEEE VTC, 2005. 19

[40] J. Leguay, T. Friedman, and V. Conan. Evaluating mobility pattern space routing for DTNs. In Proceedings of IEEE INFOCOM, 2006. 19

[41] M. Musolesi, S. Hailes, and C. Mascolo. Adaptive routing for intermittently connected mobile ad hoc networks. In Proceedings of IEEE WoWMoM, 2005. 21, 38

[42] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli. Age matters: efficient route discovery in mobile ad hoc networks using encounter ages. In Proceedings of ACM/IEEE MOBIHOC, 2003. 20

[43] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. SIGMOBILE Mobile Computing and Communication Review, 7(3):19–20, 2003. 17, 20, 21, 23, 24, 35, 38, 46, 73

[44] X. Chen and A. L. Murphy. Enabling disconnected transitive communication in mobile ad hoc networks. In Proceedings of ACM POMC, 2001. 23

[45] D. Aldous and J. Fill. Reversible markov chains and random walks on graphs. http://stat-www.berkeley.edu/users/aldous/RWG/book.html 24, 44, 115

[46] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Efficient Routing in Intermittently Connected Mobile Networks: The Single-copy Case. ACM/IEEE journal of Transactions on Networking vol. 16,NO. 1, Feb. 2008 38

[47] K. Lee, Y. Yi, J. Jeong, H. Won, I. Rhee, and S. Chong. Max-Contribution: On optimal resource allocation in delay tolerant networks. In Proceeding of INFOCOM, 2010. 20

[48] M. McNett and G. M. Voelker. Access and mobility of wireless pda users. ACM Mobile Computing and Communication Review, 2003. 44

[49] A. Chaintreau, P. Hui, J. Scott, R. Gass, J. Crowcroft, and C. Diot. Impact of human mobility on opportunistic forwarding algorithms. IEEE Transactions on Mobile Computing, 6(6):606-620, June 2007. 27

[50] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Performance analysis of mobility-assisted routing. In Proceedings of ACM/IEEE MOBIHOC, 2006. 43, 44, 45, 53, 76, 81, 94, 96, 115, 129

[51] "DTN java simulator," http://people.ee.ethz.ch/ spyropot/dtnsim.html. 53, 94

[52] T. Spyropoulos, K. Psounis, and C. S. Raghavendra.Efficient Routing in Intermittently Connected Mobile Networks: The Multi-copy Case. ACM/IEEE journal of Transactions on Networking vol. 16, NO. 1, Feb. 2008 16, 30, 41, 53, 55, 56, 75, 95

[53] W. Zhao, Y. Chen, M. Ammar, M. Corner, B. Levine and E. Zegura. Capacity enhancement using throw boxes in DTNs. In Proceedings of IEEE MASS, 2006. 37

[54] T. Spyropoulos, T. Turletti, and K. Obrazcka. Routing in Delay Tolerant networks Comprising Heterogeneous Populations of Nodes. IEEE TMC, vol8 no. 8, pp 1132-1147, August 2009. 11, 36, 37, 56, 112

[55] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Spray and wait: Efficient routing in intermittently connected mobile networks. In Proceedings of ACM SIGCOMM, 2005. 17

[56] A. Nandan, S. Das, B. Zhou, G. Pau, and M. Gerla. AdTorrent: digital billboards for vehicular networks. In Proceedings of IEEE/ACM V2VCOM, 2005. 37

[57] B. Karp and H. Kung. Greedy perimeter stateless routing for wireless networks. In Proceedings of ACM/IEEE MobiCom, 2000. 37

[58] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility. In Proceedings of IEEE PERCOM, 2007. 23, 25

[59] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Utility-based Message Replication for Intermittently Connected Heterogeneous Wireless Networks. In Proceedings of IEEE WoWMoM workshop on Autonomic and Opportunistic Communications (AOC), (INRIA Technical Report RR-6129), 2007. 20

[60] P. Gupta and P. R. Kumar. Critical power for asymptotic connectivity in wireless networks. In Stochastic Analysis, Control, Optimization and Applications, 1998. 38, 43

[61] Z. Li and Haiying Shen. Utility-based Distributed Routing in Intermittently Connected Networks. 37th International Conference on Parallel Processing. (ICPP '08), 2008. 22

[62] V. Erramilli, M. Crovella, A. Chaintreau , C. Diot. Delegation forwarding. Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing, 2008. 20, 38, 95

[63] A. Elwhishi and P. Ho. SARP - a novel multi-copy routing protocol for intermittently connected mobile networks. In Proceedings of GLOBECOM, 2009. 39, 56, 92, 93, 95, 155

[64] A. Elwhishi, Pin-Han Ho, K. Naik, and B. Shihadda. ARBR: Adaptive Reinforcement-Based Routing for DTN. In Proceedings of IEEE WiMob, 2010. 39, 95, 155

[65] M. Musolesi, and C. Mascolo. CAR: Context-aware Adaptive Routing for Delay Tolerant Mobile Networks. IEEE Transactions on Mobile Computing. 8, 2 (February 2009), 246-260. 21, 28

[66] M. V. Thomas Karagiannis, Jean-Yves Le Boudec. Power law and exponential decay of inter contact times between mobile devices. In Proceeding of ACM/IEEE MobiCom, 2007. 76, 115, 140

[67] E. Jones, L. Lily , and P. Ward. Practical Routing in Delay-Tolerant Networks. In SIGCOMM'05 Workshops, 2005. 73

[68] Q. Yuan, I. Cardei, and J. Wu. 2009. Predict and relay: an efficient routing in disruption-tolerant networks. In Proceedings of ACM MobiHoc, 2009. 20

[69] A. Balasubramanian, B. Levine, and A. Venkataramani. Replication routing in DTNs: A resource allocation approach. IEEE/ACM Transactions on Networking 18(2): 596–609, 2009. 20, 128, 148

[70] Shen Ling, Wu Wei. Feedback Adaptive Routing Algorithm for DTN. CMC, vol. 2, pp.267-271, 2009 WRI International Conference on Communications and Mobile Computing, 2009. 20

[71] V. Erramilli, M. Crovella. Forwarding in opportunistic Networks with Resource constraints. In Proceedings of ACM CHANTS, 2008. 20

[72] Demmer, M., Fall, K. Delay Tolerant Routing for Developing Regions. In Proceedings of the workshop on Networked systems for developing regions (NSDR '07). 2

[73] M. Akkouchi. On the convolution of exponential distributions. J. Chungcheong Math. Soc, vol. 21 no.4. pp.501–510, 2008. 77

[74] A. Jindal and K. Psounis. Contention-aware performance analysis of mobility-assisted routing. IEEE Transactions on Mobile Computing, v.8 n.2, p.145-161, February 2009 . 83

[75] J. Dowling, E Curran, R. Cunningham, V. Cahill. Using Feedback in Collaborative Reinforcement Learning to Adaptively Optimise MANET Routing, IEEE Transactions on Systems, Man and Cybernetics (Part A), Special Issue on Engineering Self-Orangized Distributed Systems, pp. 360-372, Vol. 35, No. 3, ISSN 0018-9472, 2005. 28, 30, 84

[76] A. Krifa, C. Barakat, and T. Spyropoulos. Optimal buffer management policies for delay tolerant networks. In Proceeding of IEEE SECON, 2008. 32, 128

[77] A. Lindgren and K. Phanse. Evaluation of Queueing Policies and Forwarding Strategies for Routing in Intermittently Connected Networks. First International Conference on Communication System Software and Middleware, 2006. 31, 128

[78] J. Dowling, R. Cunningham, E. Curran, and V. Cahill. Building autonomic systems using collaborative reinforcement learning. Knowl. Eng. Rev., 21(3):231–238, 2006. 84

[79] J. Dowling and S. Haridi. Decentralized Reinforcement Learning for the Online Optimization of Distributed Systems. Reinforcement Learning: Theory and Applications, Advanced Robotic Systems Journal I-Tech Education and Publishing, ISBN 978-3-902613-14-1, 2008: 142-167. 28, 89

[80] S. C. Nelson, M. Bakht, R. Kravets, and A. Harris. Encounter based routing in DTNs. ACM SIGMOBILE Mobile Computing and Communications Review, Vol. 13, No. 1. (June 2009), pp. 56-59. 84, 128, 148

[81] Y.-K. Ip, W.-C. Lau and O.-C. Yue. Performance modeling of epidemic routing with heterogeneous node types. In Proceeding of IEEE ICC, 2008. 111

[82] A. Elwhishi, P-H. Ho, K. Naik, and B. Shihada. A Novel Buffer Management Architecture for Epidemic Routing in Delay Tolerant Networks. In Proceeding of the QShine, 2010. (Invited paper, 8 pages). 42, 128, 156

[83] S. Dimitriou, V. Tsaoussidis: Effective buffer and storage management in DTN nodes. ICUMT 2009. 32

[84] R. S D. Wahidabanu and G.Fathima. Article: A New Queuing Policy for Delay Tolerant Networks. International Journal of Computer Applications 1(20):52–55, February 2010. 32

[85] A. Elwhishi, P-H. Ho, K. Naik, and B. Shihada. Contention Aware Routing for Intermittently Connected Mobile Networks. In Proceeding of the International Conference on Advances in Future Internet (AFIN), 2011. (Received Best Paper Award). 39, 42, 95, 139, 155

[86] J. Hillston. Fluid Flow Approximation of PEPA models. In Proceedings of QEST, 2005). 111

[87] Y. Wang, P. Zhang, T. Liu, C. Sadler and M. Martonosi. Movement Data Traces from Princeton ZebraNet Deployments. CRAWDAD Database. 129

[88] E. Daly, M. Haahr. Social network analysis for information flow in disconnected delay-dolerant MANETs. IEEE Transactions on Mobile Computing 8(5), pp 606–621, 2005. 27

[89] J. Freyne, S. Berkovsky, E. Daly, and W. Geyer. Social networking feeds: recommending items of interest. In Proceedings of ACM RecSys, 2010. 27

[90] P. Hui, J. Crowcroft, and E. Yoneki. Bubble Rap: Social-based forwarding in delay tolerant networks. In Proceedings of ACM MobiHoc, 2008. 27

[91] T. Abdelkader, K. Naik, A. Nayak, and N. Goel. A Socially-Based Routing Protocol for Delay Tolerant Networks. In Proceedings of IEEE GLOBECOM, 2010. 27

[92] "CRAWDAD: A Community Resource for Archiving Wireless Data at Dartmouth," http://crawdad.cs.dartmouth.edu/. 129

[93] P. Costa, C. Mascolo, M. Musolesi, G. Picco. Socially-Aware Routing for Publish-Subscribe in Delay-Tolerant Mobile Ad Hoc Networks. IEEE Journal on Selected Areas in Communications 26 (2008) 748-760. 28

[94] P. Hui, E. Yoneki, J. Crowcroft, S. yan Chan, S. Identifying Social Communities in Complex Communications for Network Eciency. In Proceedings of Complex, 2009. 27

[95] J. Lakkakorpi, M. Pitkänen, and J. Ott. Using buffer space advertisements to avoid congestion in mobile opportunistic DTNs. In Proceedings of WWIC, 2011. 30

[96] J. Link, D. Schmitz, K. Wehrle. GeoDTN: Geographic Routing in Disruption Tolerant Networks. In Proceedings of IEEE GLOBECOM, 2011. 19

[97] K. A. Harras and K. C. Almeroth. Transport layer issues in delay tolerant mobile networks. In Proceedings of IFIP-TC6 Networking, 2006. 35

[98] T. Spyropoulos, T. R. N. B. Rais , T. Turletti , K. Obraczka, A. Vasilakos. Routing for disruption tolerant networks: Taxonomy and design. Wireless Networks 16(8), pp. 2349–2370, 2010. 30

[99] Tang L., Zheng Q., Liu J., Hong X. Selective message forwarding in delay tolerant networks. Mobile Networks and Applications 14, 2009, pp. 387–400. 30

[100] J. Zhang, L. Guangchun. Adaptive Spraying for Routing in Delay Tolerant Networks. Wireless Personal Communications. Springer, 2011. 17

[101] P. Mundur, M. Seligman, and G. Lee. Epidemic routing with immunity in Delay Tolerant Networks. In Proceedings of the Military Communications Conference, 2008. 17

[102] S. Lo, M. Chiang, J. Liou, and J. Gao. Routing and Buffering Strategies in Delay-Tolerant Networks: Survey and Evaluation. In Proceedings of ICPP Workshops, 2011. 38, 41

[103] A. Elwhishi, P-H. Ho, K. Naik, and B. Shihada. Self Adaptive Contention Aware Routing Protocol for Intermittently Connected Mobile Networks. To appear in IEEE Transactions Parallel Distributed Systems, 2012. 155

[104] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Impact of Human Mobility on Opportunistic Forwarding Algorithms. IEEE Trans. on Mobile Computing 6, 6 (2007), 606–620.

94, 107, 149