Transport Layer Issues in Delay Tolerant Mobile Networks

Khaled A. Harras and Kevin C. Almeroth

Department of Computer Science
University of California, Santa Barbara
Santa Barbara, CA 93106-5110
{kharras, almeroth}@cs.ucsb.edu

Abstract. The tremendous increase in wireless devices and user mobility have ultimately resulted in a new set of networking challenges that previously did not exist. Some of these challenges include large delays, intermittent connectivity and most importantly, the absence of an end-to-end path from sources to destinations. Networks characterized by one or more of these challenges are called *Delay Tolerant Networks (DTNs)*. Researchers have studied DTNs with a major focus on routing issues in such extreme environments. As a result, in this paper, we shift this focus towards addressing and studying transport layer issues in extreme networking environments. We particularly concentrate on investigating and comparing several reliability approaches in a specific category of DTNs known as Delay Tolerant Mobile Networks (DTMNs). We present four different reliability approaches in DTMNs. We also evaluate these approaches under various network conditions via simulation. Our goals from this study are to examine the impact of these reliability approaches, understand the tradeoffs between them, and open the way for further work in transport layer issues in delay tolerant networks.

Keywords: Delay Tolerant Networks, Mobile Networks, Reliability

1 Introduction

With the explosive evolution in wireless devices, many new network environments have emerged. Some of these environments include, satellite and interplanetary [7], military/tactical [11], disconnected remote village [13], and disaster rescue [1] networks. These new environments have become more prominent with recent natural disasters. The need to establish communication to serve applications that run in such extreme environments has never been more evident.

The emergence of these new environments has lead to a new set of networking challenges. Some of these challenges include network partitioning, large delays, intermittent connectivity, high link error rates, and heterogeneous underlying networks and protocols. As a result, a new set of assumptions needs to be considered, such as large delays, intermittent connectivity, and most importantly, the absence of an end-to-end path from a source to a destination.

These new challenges and assumptions have spurred much research in such extreme and mobile environments. Researchers in Mobile Ad Hoc NETworks (MANETs) have tackled mobility problems with a major focus on routing [9], [14], [15], [16]. MANETs, however, fail to address all of the emerging challenges listed above, since they only consider scenarios where an end-to-end path exists

from a source to a destination. Other research has started to address the challenge of communicating even though such a path does not exist. This research includes disconnected mobile networks [12], [18], sparse sensor networks [10], [17], and different forms of Delay Tolerant Networks (DTNs) [5], [4], [8], [19], [20], [6]. These areas have introduced different DTN architectures and solutions with a focus on solving *routing* and *message delivery* problems in such extreme environments.

With the work in DTNs mainly focused on routing, we shift our focus towards studying transport layer issues. Most of the services offered by existing transport layer protocols, such as TCP, have been overlooked. In general, the most important services offered by TCP are ports, connections, sequencing, congestion control, and reliability. Some of these services are easy to deploy in DTNs, while others require further research. We briefly look at each of the TCP-style transport functions in DTN environments.

Of the TCP services previously mentioned, ports are still provided and used by overlay protocols for communication in DTN environments. Next, sequencing is done the same way as in TCP, with the exception that sequence numbers are assigned to *message bundles* rather than to individual packets. Connection establishment, on the other hand, is impossible in such environments due to the primary assumption of the absence of an end-to-end connection. The only remaining services, therefore, are congestion control and reliability. Congestion control is a more challenging function to deploy because propagating live congestion-related information across DTN environments is hard. This difficulty is due to the unstable nature of DTN environments. Addressing congestion control is left to future work. We are now left to focus on reliability, a service critical to many of the applications that run in DTN environments.

In this paper, we introduce four different end-to-end reliability approaches for a specific DTN architecture, known as Delay Tolerant Mobile Networks (DTMNs), which are large-scale disconnected mobile networks [6]. First, *hop-by-hop* reliability depends only on sending acknowledgments along every hop in the path. Second, *active receipt* achieves reliability by delivering an *active* end-to-end acknowledgment over the DTMN. Third, *passive receipt* reliability implicitly sends an end-to-end acknowledgment through the network. Fourth, *network-bridged receipt* sends an acknowledgment over another network that exists in parallel to the DTMN. With the multiple devices people currently carry, we can use other parallel networks, such as cell networks, as network bridges to transmit acknowledgements or other control-related information. We evaluate these reliability approaches in DTMNs under various network conditions via simulations. Our goals in this study are to examine the impact of these reliability approaches, understand the tradeoffs between them, and open the way for further work in transport layer issues in delay tolerant networks.

The remainder of this paper is organized as follows. Section 2 first introduces related work. Section 3 then gives an overview of DTMNs. We discuss the different reliability approaches in Section 4. The simulation environment and results are described in Section 5. Finally, we conclude in Section 6.

2 Related Work

Research in the areas of MANETs [9], [14], [15], [16], disconnected mobile networks [12], [20], [19], sparse sensor networks [17], and delay tolerant networks (DTNs) [4], [8], [6], have addressed issues related to the challenges stated in Section 1. We briefly present some of the solutions in these areas.

Work in MANETs has mainly focused on routing, introducing various protocols that find end-to-end paths between nodes [9], [14], [15], [16]. Since such paths mostly do not exist in the applications with which we are concerned, MANETs, therefore, fail to address the transport challenges we address in this paper.

Most of the solutions presented by disconnected mobile and sparse sensor networks rely on some form of store-and-forward relaying of messages. This relaying includes different message delivery techniques; the differences are in the underlying assumptions over which they operate. For example, some solutions assume full control over node movement [12]. Others, such as message ferrying, assume knowing the path that some nodes will take and the time at which these node will take that path [20]. Some consider using *data mules* to gather data from static sensors [17], while others find optimal paths for *ferries* to deliver messages between sparse static nodes [19]. Epidemic Routing, on the other hand, simply floods the network to ensure message delivery [18]. Our previous work provides different approaches to control these floods [6]. All of these solutions fundamentally focus on message delivery and routing techniques in challenged extreme environments. To the best of our knowledge, no existing work thoroughly studies transport layer issues in such environments.

With respect to DTNs, members of the Delay Tolerant Networking Research Group (DTNRG) [3] introduce an architecture that helps achieve connectivity among heterogeneous networks in extreme environments [2], [4]. A *bundle layer protocol* is introduced to handle many of the challenges previously discussed using a store-and-forward approach. They also propose the idea of *custody transfer*, where a *custodian* assumes the responsibility of reliably delivering a bundle to the next custodian on the path to the destination [5]. Jain et al. expand on the DTN work by studying routing issues in such extreme environments. Again, the focus in the DTN work is almost exclusively on routing [8]. The DTN community has briefly addressed reliability through custody transfer [5] in the bundle layer protocol [4]. However, there has been no in-depth study or evaluation of its performance, especially when compared to other approaches. In this paper, we examine this approach, along with others that we propose, particularly over delay tolerant mobile networks (DTMNs).

3 An Overview on DTMNs

The work presented in this paper uses DTMNs [6] as the underlying network environment. Since this is the environment we use to study our reliability approaches, we give a brief overview of DTMNs' basic architecture and terminology.

DTMNs are a special kind of DTNs with the assumption that all nodes in the network are mobile, and that end-to-end paths may not exist between any two nodes in the network. In this environment, due to the sparseness and

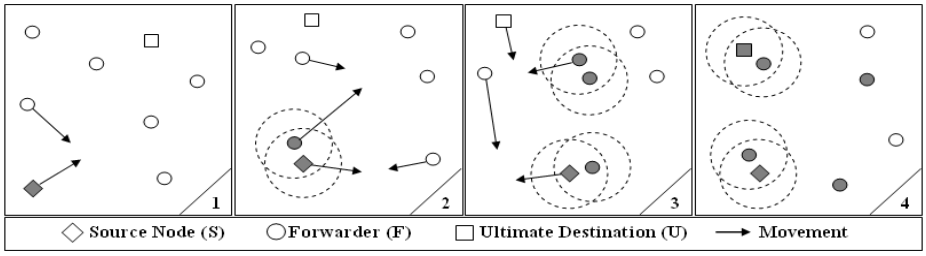


Fig. 1. An example of message delivery in DTMNs. *Infected* nodes are shaded.

mobility of nodes, each node is viewed as a "region" with respect to the classical DTN architecture [4]. Similarly, each node acts as a DTN gateway to perform overlay bundle relaying of messages. There are two key assumptions in DTMNs with respect to network nodes. First, nodes are *blind*. They do not know any information regarding the state, location, or mobility patterns of other nodes. Second, nodes are *autonomous*. Each node has independent control over itself and its movement.

We now show the operation of DTMNs, an example of which is illustrated in Figure 1. The number in the bottom right corner of each sub-figure represents the sequence of snapshots taken for a DTMN. The figure shows the basic method for propagating messages through the network from the source node, S, to the ultimate destination, U, with the aid of other forwarder nodes, F. Shaded nodes are what we refer to as *infected nodes*, nodes which have received a copy of the message. All infected nodes, including the source, try to infect other nodes at varying degrees of *willingness*. This willingness is generally an indication of how hard a given node tries to forward to, or infect, other nodes.

We note that a DTMN could be viewed both as a full DTN in itself, where each node is both a region and a DTN gateway, or as a single region within the classical DTN architecture [4]. Due to this vagueness, we study the reliability approaches only over DTMN environments in order to focus on the performance and tradeoffs between these approaches. We believe, however, that the results of our work will help us better understand reliability challenges in DTNs in general.

4 Reliability Approaches

We present in this section the four reliability approaches that we study in this paper. First, we discuss the most basic reliability approach for DTMNs, which is *hop-by-hop*. Afterwards, we talk about two different approaches for delivering an end-to-end acknowledgement over a DTMN. These approaches are *active receipt* and *passive receipt*. Finally, we propose a novel modification to the typical DTMN architecture by introducing the idea of *network-bridged receipt*.

4.1 Hop-by-Hop

Hop-by-hop reliability was first introduced in classical DTNs [4]. The idea there, however, was to deliver a message across a given region on the path to the destination, where each region represents a hop. Gateways at the edges of these re-

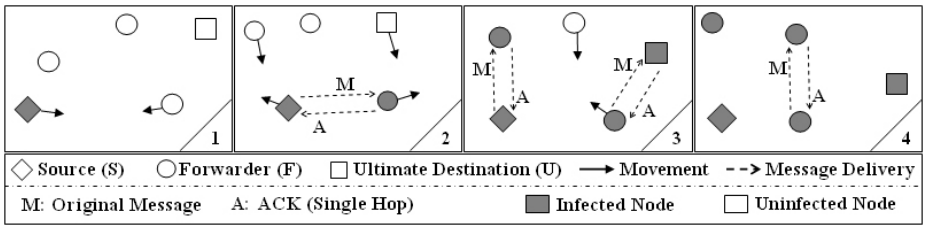


Fig. 2. The operation of hop-by-hop reliability in DTMNs.

gions act as custodians and take the responsibility of reliably delivering message bundles across the region [5]. Therefore, there is no end-to-end acknowledgment in these cases; the source only knows whether the next gateway received the message or not, and assumes the gateway will take care of the rest. We build on this idea, and use it as the base reliability approach for DTMNs.

We apply hop-by-hop reliability, however, differently in DTMNs. With the extreme hostility and mobility assumed in DTMN applications, each node in the network acts as a region *and* a gateway with respect to the DTN architecture. Therefore, any exchange of messages between nodes is acknowledged, and all nodes are assumed to reliably forward the message.

The operation of hop-by-hop reliability in DTMNs is illustrated in Figure 2. The source, S, sends a message, M, to the ultimate destination, U, with the aid of forwarder nodes, F. Each time M is *successfully* delivered to any node, an acknowledgment, A, is then sent back to acknowledge the receipt of M. The forwarder nodes along with the source node try to infect as many nodes as possible according to their willingness level. Given enough time and mobility, S assumes that M will eventually reach U. Even though hop-by-hop does not ensure end-to-end reliability, it has the advantage of minimizing the amount of time M remains in S's buffer. This is because S does not need to wait for any end-to-end acknowledgment. We use hop-by-hop as the base approach over which we build the other end-to-end reliability approaches.

4.2 Active Receipt

While the hop-by-hop approach ensures some level of reliability, it does not ensure end-to-end reliability. This limitation could be a problem in cases where failures, such as the destruction of a node in a battlefield, or the breakdown of a node in a disaster rescue operation, are likely to occur. In such cases, some form of added end-to-end reliability is required. We overcome this drawback of the hop-by-hop approach by introducing the *active receipt*.

Active receipt is basically an end-to-end acknowledgment, which we call a *receipt*, created by U after it receives M from S. This receipt is *actively* sent back to S. By "actively", we mean that nodes treat this receipt as a new message that needs to be forwarded.

We demonstrate the operation of active receipt in Figure 3(a). The first snapshot starts at the time when U has just received M, with most of the F nodes already infected with M. U then creates the active receipt, R, which is

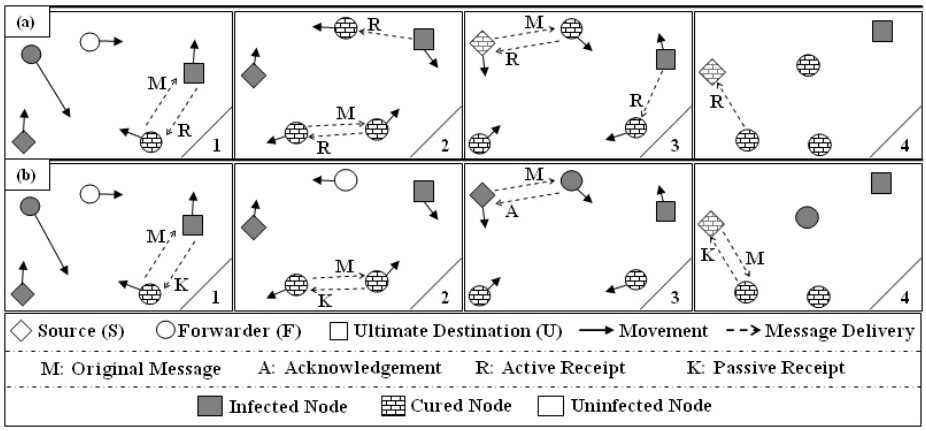


Fig. 3. Demonstrating and comparing (a) active receipt and (b) passive receipt reliability approaches in DTMNs.

forwarded through the F nodes until it reaches S, shown in the third snapshot of Figure 3(a). Throughout this process, we observe how R *cures* the infected nodes in the network by stopping their transmission of M. R is also cached according to the nodes' willingness levels to prevent re-infection of M. Even though this cure eventually stops the epidemic spread of M through the network, R itself starts to spread epidemically until some timeout or TTL value. The cost of carrying and transmitting R, however, is less than M due to the small size of R.

4.3 Passive Receipt

While active receipt offers end-to-end reliability, its cost in many situations is high. This high cost is because active receipt reaches a point where two messages, rather than one, are infecting nodes in the network. Therefore, we introduce *passive receipt*, which ensures end-to-end reliability, without the incurred cost of active receipt. The idea is to have an implicit/passive receipt, instead of an active one, traverse the network back to S.

We use Figure 3(b) to help clarify the operation of passive receipt. The first snapshot, similar to Figure 3(a), starts at the time when U just received M. However, instead of generating a new active receipt, R, an implicit kill message, K, is sent to the infected node to stop it from sending M. The idea is that K is sent by the cured nodes (or U) *only* when they are encountered by one of the infected nodes trying to pass M on to them. In other words, cured nodes do not actively send K messages, they simply wait for active infected nodes to come in their way and stop them from sending M.

The operation of the passive receipt is better understood when compared to active receipt, as illustrated in Figure 3. The first difference is shown in both second snapshots, where in Figure 3(a), R is actively sent to an infected as well as an uninfected node. In the case of passive receipt shown in second snapshot of Figure 3(b), however, K is only sent to the infected node *after* this infected node had tried to pass M to a cured node.

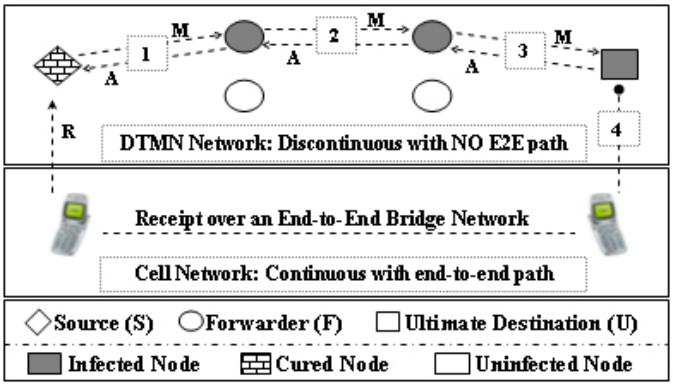


Fig. 4. The network-bridged receipt reliability approach.

This reduction in cost introduced by the passive receipt approach is not free when compared to active receipt. Even though an end-to-end receipt is received by S in both cases, S receives the end-to-end receipt more rapidly in the case of active receipt. When using the passive receipt, K is received by S at the fourth snapshot, as opposed to receiving R at the third snapshot using active receipt. The reason for this difference in receipt arrival time is that with the active approach, R spreads rapidly in the network, which helps it reach S more quickly than the passively spreading K. This passiveness also results in having infected nodes in the network take a longer time to be cured, as shown in the fourth snapshot in Figure 3(b). This means that the chances of having some infected nodes still trying to send M after S received a receipt, is higher in the passive receipt approach than the active receipt.

4.4 Network-Bridged Receipt

We now introduce a new assumption to the DTMN architecture that enables us to create another reliability approach. This assumption is based on the widespread use of cell phones. We propose exploiting the availability of the cell network by using it as an alternative path for our communication protocol. While such a network does not have the required bandwidth for delivering large amounts of data, it *could be* used for transmitting lightweight control information. Therefore, we use this cell network only for transmitting an end-to-end receipt from the destination back to the source.

This idea is illustrated in Figure 4. We note that all nodes in the network are capable of mobility, however, for clarity, we do not include mobility in the figure. The cell network acts as a bridge between nodes in the DTMN. The cell network is characterized by its continuous end-to-end, low bandwidth connections. The DTMN network, on the other hand, is characterized by its discontinuous non-end-to-end, high bandwidth. In such a setup, large messages, M, are typically transmitted from S over the DTMN using the base hop-by-hop reliability approach until it reaches U. The end-to-end network-bridged receipt, R, would then be transmitted over the cell network instead of the DTMN. If we assume

that other nodes in the network also have access to the cell network, R could then be transmitted to these nodes. The result is a very rapid cure for all infected nodes in the network.

The advantage of the network-bridged approach is to reduce the round trip time between nodes S and U roughly by half. Consequently, the message is dropped faster from the queue in A since the receipt arrives faster. The drawback, however, lies in the assumption itself: the added complexity of bridging the DTMN network with the cell network. We believe, however, that the interconnection of these two networks is a likely possibility in the future.

5 Evaluation

The primary goal of our evaluation is to compare the performance and examine the tradeoffs between the reliability approaches described in Section 4. We first describe our simulation setup and environment. We then summarize the outcomes of an extended set of simulations we conducted. The extended result set is not shown due to space limitation. Therefore, we only present a subset of our results that most clearly allows us to show the tradeoffs between our reliability approaches.

5.1 Simulation Environment

We conducted our simulations using the GloMoSim network simulator. We added an overlay layer that handles message bundle relaying and implements the reliability approaches that we have described. We use a *modified* random way-point mobility model that avoids the major problem of node slow down in the conventional random way-point model. We believe this model closely approximates the scenarios with which we are concerned, such as battlefields or disaster rescue operations, due to their hostility and unpredictable movement. The node speed ranges between 20 to 35 meters per second, and the rest period is between 0 and 10 seconds. We examined other ranges as well, and they produced similar results with respect to our reliability approaches. Every point in our results is taken as an average of ten different seeds.

The major parameters used in our simulations are summarized in Table 1. The *Terrain* is the area over which the *Number of Nodes* are scattered. *Simulated Time* represents the amount of time the simulations run. The *Beacon Interval* is the period after which beacons are sent. A "beacon" is simply a signal emitted by all nodes to search for other nodes in the network as well as to announce its location. The *Times-To-Send* (TTS) is the number of times a node will successfully forward a message to other nodes in the network. *Retransmission Wait Time* represents the amount of time a node remains idle after successfully forwarding a message to another node. When the retransmission wait time expires, the node then tries to resend the same message. We mainly use TTS to represent the *willingness* of the nodes to participate in message relaying. Finally, the *reliability approach* parameter represents our four different acknowledgement schemes.

We consider three main metrics in evaluating our reliability approaches. The first metric is *Cost*, which is the total number of messages sent by all nodes in

Table 1. Simulation Parameters

Parameter	Value Range	Nominal Value
Terrain	$10km^2$ to $50km^2$	$10km^2$
Number of Nodes	10 to 250	100
Simulated Time	1hour to 24 hours	6 hours
Beacon Interval	0.5sec to 50sec	1sec
Times-To-Send	1 to 50	10
Retransmission Wait Time	0sec to 500sec	50sec
Reliability Approach	Hop-by-hop, Active, Passive or Network-Bridged	N/A

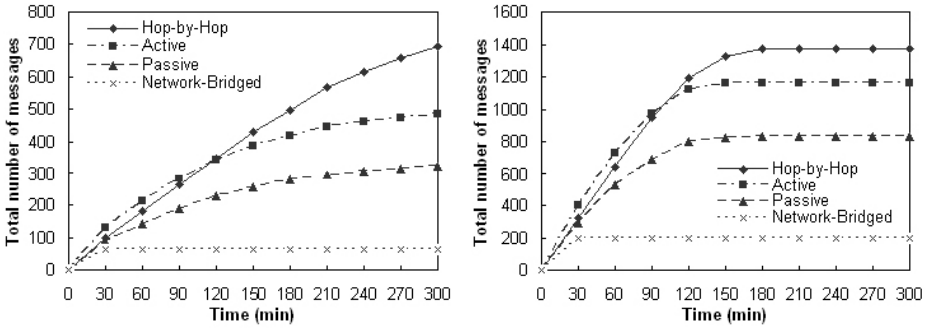
the network. The second metric is *Queuing Time*, which is the average time a message remains in the sender node's queue before it is dropped. The third metric we consider is *Delivery Ratio*, which is the percentage of messages delivered. We choose to focus on the first two metrics since delivery ratios in DTMNs simply depend on the time ceiling set for message delivery, i.e. given enough time, all messages will eventually be delivered.

5.2 Results

We present a summary of the extended set of simulations, along with a subset of our simulation results, which clarify and support our conclusions. All the results are shown for a single sender node sending one message to a single ultimate destination. The purpose of our simulations is twofold. First, we hope to better understand how different reliability approaches behave when run in a DTMN. Second, we want to understand the tradeoffs between these approaches.

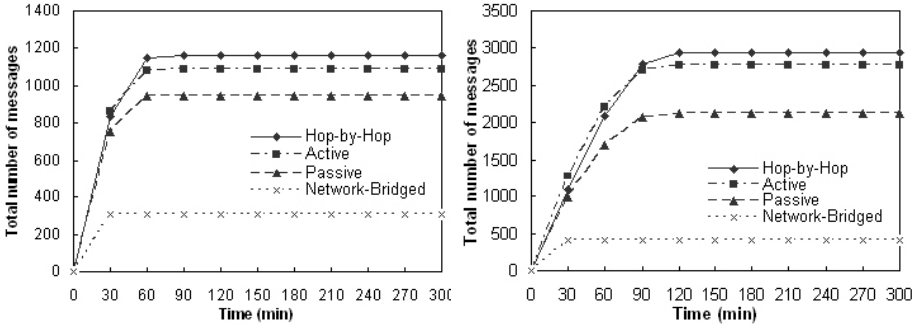
Generally speaking, the network-bridged receipt incurs the least cost when compared to the other approaches. The highest cost, on the other hand, occurs with the hop-by-hop approach. The cost of the active and passive receipts fall in between, with active receipt being relatively more expensive. These observations are supported by Figure 5 and Figure 6, which demonstrates the cost of each reliability approach in terms of the total number of messages sent. We measure this cost under different network densities, 25 nodes in Figure 5(a), 50 nodes in Figure 5(b) and 100 nodes in Figure 6(d), as well as different willingness levels, times-to-send is set to 5 in Figure 6(a) and 10 in Figure 6(b). One interesting observation is where the cost of the active receipt is the highest until it is eventually exceeded by the hop-by-hop approach. This result is because after the message reaches the ultimate destination, we now have two messages infecting the network, which creates this large cost. Eventually, however, the receipt cures those nodes infected with the original message and is itself cured after reaching the source node. We note also that changes in node density or willingness levels have minor impact on the *relative* performance of our reliability approaches.

Even though the performance of the reliability approaches is relatively similar over different network densities, other aspects, such as the rate of message spreading and convergence, vary. This result is particularly evident in the difference in the Y-axis scales of Figure 5 and Figure 6. Generally speaking, the



(a) 25 nodes (b) 50 nodes

Fig. 5. The cost of the reliability approaches over time in DTMNs with different node densities. Graphs (a) and (b) represent 25 and 50 nodes, both with a TTS of 10.



(a) Times to send = 5 (b) Times to send = 10

Fig. 6. The cost of the reliability approaches over time in DTMNs with different willingness levels. Graphs (a) and (b) represent TTS of 5 and 10, both with 100 nodes.

messages spread faster in denser networks. This observation can be seen by the sharper increase in the total number of messages in the case of Figure 6(b) when compared to Figures 5(a) and Figure 5(b). We compare Figure 6(b) with Figure 5 since the former measures the cost over a 100 node network with the same TTS value of 10 as that used in Figure 5. Alternatively, the network heals faster in denser networks. This result is shown in the faster convergence of the lines in Figure 5(b) when compared to those in Figure 5(a). This convergence leads to a steady horizontal line because the network reaches a point of saturation where it no longer needs to forward the message.

Regarding the average queuing time, the results show that the hop-by-hop approach has the lowest value. This low value is because the source node does not wait for any end-to-end acknowledgement to be received, and therefore, drops the message from its buffer after forwarding to other nodes in the network. If end-to-end reliability is required, the best approach in terms of minimal queuing time is the network-bridged approach. Figure 7 supports these observations by illustrating the average queuing time of a given message with respect to our

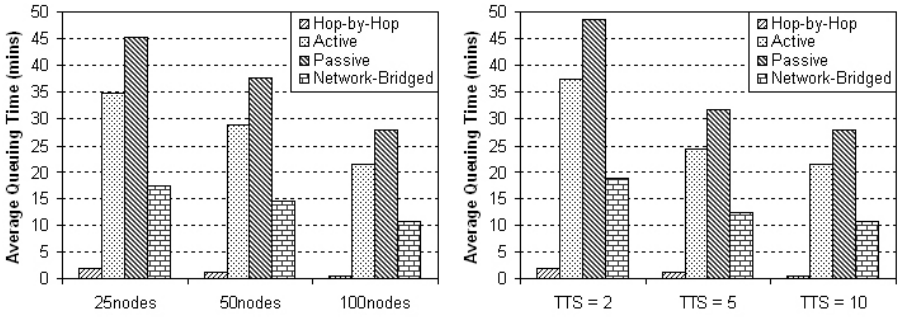


Fig. 7. The impact of (a) the number of nodes, and (b) the times-to-send on the average queuing time of a message at the sender node.

reliability approaches under (a) different densities, and (b) different willingness levels. The other interesting fact Figure 7 highlights, is that active receipt has less queuing time than passive receipt. This fact offers a tradeoff for the extra cost incurred in the active receipt when compared to the passive receipt approach. The reason for this result is due to the active way in which the receipt is sent when compared to the passive approach. The active approach results in the receipt reaching the source faster, but at a higher cost.

Figure 7 also shows that the tradeoffs between the reliability approaches is generally similar over different densities and different willingness levels. The primary difference is that the overall queuing time of all the reliability approaches decreases as the network density or willingness levels increase. This result is because in denser networks, or when nodes are trying harder to forward a message, the overall end-to-end delay decreases. This decrease in delay consequently leads to smaller queuing time.

6 Conclusions and Future Work

In this paper, we have considered transport layer issues, specifically reliability, over a special class of DTNs knows as DTMNs. We introduced four different reliability approaches: hop-by-hop, active receipt, passive receipt, and network-bridged receipt. We have investigated and evaluated these approaches via simulation. Overall, we discovered that the choice of the most suitable reliability approach depends on the expected complexity of the underlying DTMN. For example, the hop-by-hop is the simplest, while network-bridged is the most complex. Also, the priority of cost versus delay governs the choice between the active and passive receipt.

We consider this paper a next step in thoroughly investigating transport layer issues in DTNs in general. Our future work, therefore, is to apply these approaches to DTNs in general, and see how they might be modified and applied to other DTN architectures. Also, we intend to address other transport layer issues, particularly, congestion control.

References

1. University of South Florida: Center for robot-assisted search and rescue. http://crasar.csee.usf.edu/.
2. V. Cerf, et. al. Interplanetary Internet (IPN): Architectural Definition. *IETF Internet Draft, draft-irtf-ipnrg-arch-00.txt*, May 2001.
3. DTNRG. Delay Tolerant Networking Research Group. http://www.dtnrg.org/.
4. K. Fall. A Delay-Tolerant Network Architecture for Challenged Internets. In *ACM SIGCOMM*, Karlsruhe, Germany, August 2003.
5. K. Fall, W. Hong, and S. Madden. Custody Transfer for Reliable Delivery in Delay Tolerant Networks. *Intel Research, Berkeley-TR-03-030*, July 2003.
6. K. Harras, K. Almeroth, and E. Belding-Royer. Delay Tolerant Mobile Networks (DTMNs): Controlled Flooding Schemes in Sparse Mobile Networks. In *IFIP Networking*, Waterloo, Canada, May 2005.
7. A. Hooke. The Interplanetary Internet. *Communications of the ACM*, 44(9):38–40, September 2001.
8. S. Jain, K. Fall, and R. Patra. Routing in a Delay Tolerant Network. In *ACM SIGCOMM*, Portland, OR, August 2004.
9. D. Johnson and D. Maltz. *Dynamic Source Routing in Ad Hoc Wireless Networks*, volume 353. Kluwer Academic Publishers, 1996.
10. P. Juang, et. al. Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences With ZebraNet. In *In International Conference on Architectural Support for Programming Languages and Operating Systems*, San Jose, CA, October 2002.
11. E. Krotkov and J. Blitch. The Defense Advanced Research Projects Agency (DARPA) Tactical Mobile Robotics Program. *The International Journal of Robotics Research*, 18(7):769–776, July 1999.
12. Q. Li and D. Rus. Sending Messages to Mobile Users in Disconnected Ad-Hoc Wireless Networks. In *ACM MobiCom*, pages 44–55, Boston, MA, August 2000.
13. A. Pentland, R. Fletcher, and A. Hasson. Daknet: Rethinking connectivity in developing nations. *Computer*, 37(1):78–83, 2004.
14. C. Perkins. Ad-hoc On-Demand Distance Vector Routing. In *IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, New Orleans, LA, February 1999.
15. C. Perkins and P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. In *ACM SIGCOMM*, pages 234–244, London, England, October 1994.
16. E. Royer and C. Toh. A Review of Current Routing Protocols for Ad-hoc Mobile Wireless Networks. *IEEE Personal Communications Magazine*, 6(2):46–55, April 1999.
17. R. Shah, S. Roy, S. Jain, and W. Brunette. Data MULEs: Modeling a Three-Tier Architecture for Sparse Sensor Networks. In *In IEEE International Workshop on Sensor Network Protocols and Applications*, Anchorage, AK, 2003.
18. A. Vahdat and D. Becker. Epidemic Routing for Partially Connected Ad Hoc Networks. *Technical Report CS-200006, Duke University*, April 2000.
19. W. Zhao and M. Ammar and E. Zegura. Controlling the Mobility of Multiple Data Transport Ferries in a Delay-Tolerant Network. In *IEEE INFOCOM*, Miami, FL, March 2005.
20. W. Zhao, M. Ammar, and E. Zegura. A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks. In *ACM MobiHoc*, Tokyo, Japan, May 2004.