SHIM: A Scalable Hierarchical Inter-domain Multicast Approach for Disruption Tolerant Networks

Qing Ye, Liang Cheng, Mooi Choo Chuah, and Brian D. Davison
Department of Computer Science and Engineering, Lehigh University
Bethlehem, PA 18015, USA
{qiy3, cheng, chuah, davison}@cse.lehigh.edu

***Abstract*—Disruption Tolerant Network (DTN) technologies are emerging solutions to networks that experience frequent partitions. In this paper, we propose the scalable hierarchical inter-domain multicast (SHIM) approach for DTNs. SHIM has the following characteristics: *i*) it is capable of delivering multicast messages to receivers distributed in different domains; *ii*) the size of the membership information maintained by the source leader is determined by its out-degree in the leader layer, no matter how large the number of the real receivers is; and *iii*) it at least doubles the message delivery efficiency than that of directly extending the existing intra-domain DTN multicast methods to perform the inter-domain multicast operations. Our results also show that the message delivery ratio of SHIM can be improved to be almost 100% when the custodian transfer functionality is enabled in the overall networks.**

Keywords-multicasting; performance comparisons

I. INTRODUCTION AND RELATED WORK

Disruption Tolerant Networks (DTNs) [1-5] are emerging wireless networks in which the end-to-end paths between data sources and receivers are intermittently connected. Communication environments of DTNs feature frequent network partitions as the result of planned or unplanned link up/down periods between neighboring nodes. Such link layer challenges may be caused by high node mobility, low network density, limited radio ranges, scheduled node unavailability, or unexpected infrastructure disruptions. Thus, DTN routing methods are generally designed to forward messages (called bundles) in a hop-by-hop store-and-forward manner [6] between the communication peers.

DTN multicast is an important service to conduct one-to-many data communication in disruption tolerant networks. Network designers face challenges when directly applying multicast approaches proposed for the Internet or mobile ad hoc networks (MANET) to DTN environments. First, it is difficult to maintain the connectivity of a multicast structure during the lifetime of a multicast session. Second, bundle transmissions will experience a large number of failures because of the disruptions caused by repeatedly broken multicast branches. Third, most multicast approaches are designed under the assumption that the underlying networks are connected, which is hardly met in DTNs [7]. There are some existing research results that address these challenges such as OS-Multicast [8] and DTBR [9]. Both dynamically create and adjust the multicast tree according to the latest local view of the current network situations at each intermediate node.

We consider a DTN domain as a DTN sub-network where all the nodes are under the same administration. A multicast operation that only involves nodes within the same domain is called the intra-domain DTN multicast and the one that delivers bundles to receivers that are distributed in different domains is denoted as the inter-domain DTN multicast. Fig.1 shows examples of these two types of multicast operations. The traffic of the inter-domain multicast will traverse across borders of different domains. OS-Multicast and DTBR are appropriate to conduct the intra-domain multicast operation because both require the source maintains a whole list of all the receivers in certain ways. They are not scalable when there are a large number of receivers distributed different domains. Also, our results in Section III show that it is very inefficient by directly applying OS-Multicast and DTBR for the inter-domain multicast data communication.

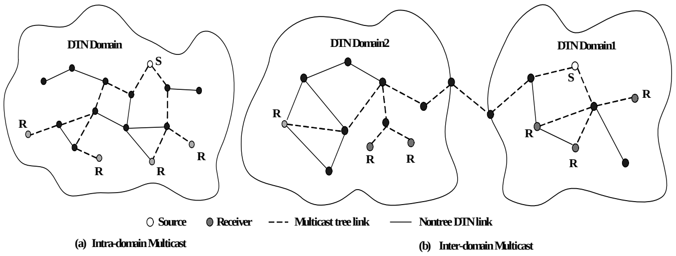


Figure 1. An example of the intra-domain and inter-domain DTN multicast.

The studies of inter-domain multicast in the Internet have been conducted in decades. The research results such as MBGP [10] and BGMP [11] combines the functionality of the intra-domain multicast approaches and BGP (Border Gateway Protocol [12]). In MBGP, each router in an autonomous system (AS) only needs to know the multicast tree within its own domain. The data communication across different domains is handled by BGP protocol. BGMP constructs an inter-domain shared tree by itself and manages the multicast addresses in a hierarchical manner. In this way, BGMP is scalable in terms of handling the multicast group information. A lot of multicast protocols have been proposed to address the challenge of the frequent topology variations in mobile wireless networks. In ODMRP [13], a multicast mesh rather than a tree is created from the source to reach the receivers that are moving in the networks. It shows that the message delivery ratio can be improved due to the nature of having multiple paths in a mesh. In [14], the features and performance of the overlay multicast in MANET is discussed and analyzed. A virtual layer consists of the source and all the receivers is built upon the real networks. Then a multicast tree is constructed from the virtual layer which adapts to the variations in the underlying networks.

In this paper, we try to combine the advantages of the above solutions with our previous experience of the OS-Multicast design and propose SHIM, a scalable hierarchical inter-domain multicast approach, to provide the inter-domain multicast service in DTNs. Compared to the flat tree constructed by OS-Multicast and DTBR, SHIM organizes the multicast structure hierarchically and effectively suppresses the management states by hiding the receiver information of the lower layer from the upper layer. Our results show that SHIM is able to achieve better message delivery efficiency than directly extending OS-Multicast or DTBR to perform inter-domain multicast operations.

The rest of this paper is organized as follows. Section II presents the details of SHIM. Section III shows our performance evaluation. And Section IV concludes this paper.

II. SCALABLE HIREARCHICAL INTER-DOMAIN MULTICAST

A. *Basic Idea and System Architecure*

We assume that each DTN domain has at least one domain leader which is a dedicated DTN node set up and managed by the domain administrator. The whole networks are then organized into two layers: the upper layer (called the *leader layer*) consisting of all the domain leaders and the multicast sources, and the lower layer (called the *domain layer*) consisting of all the other nodes within each individual domain. In SHIM, the leaders have four roles: *SL* (the source leader), *RL* (the receiver leader), *FL* (the forwarder leader), and *NL* (the normal leader). A leader that has a multicast source within its domain is called a SL and one that has at least one receiver inside its domain is called a RL. FLs are responsible of forwarding bundles from SLs to RLs in the leader level. And leaders that are not SL, RL and FL are denoted as NLs. Initially, every leader is a NL. A leader may have multiple roles. For instance, a RL is also a FL if it is an intermediate node in the multicast structure to reach some other RLs. Fig. 2 shows our hierarchical view of DTNs.

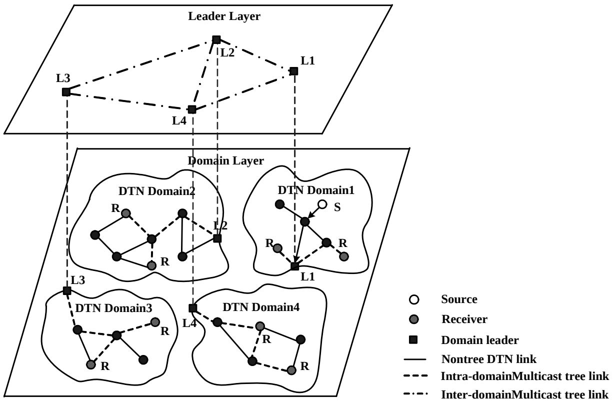


Figure 2. Two layers of the DTNs with one source and eight receivers distributed in four domains. The leader of domain1 is a source leader and also a receiver leader. The leaders of other domains are receiver leaders and may become forwarder leaders according to the variations of the multicast structure.

In SHIM, multicast bundles are transmitted in three steps: *i*) from the source to its own domain leader (a.k.a the source leader) by any DTN unicast method; *ii*) from the SL to the to the RLs by the inter-domain multicast communication in the leader layer; and *iii*) from the RLs to the real receivers by the intra-domain multicast in the domain layer.

Two leaders are neighbors in the leader layer if there is currently at least one available path connecting them in the underlying domain layer. Thus, a one-hop bundle transmission in the leader layer may actually traverse multiple DTN hops in the underlying networks. The message transmission between two neighboring leaders is conducted by the DTN unicast method supported in the underlying networks.

B. *Membership Integration*

Membership management is an important component of any multicast schemes. In SHIM, it is handled in a hierarchical way. When a multicast session starts, the source will send a *SESSION_ADV* message which is a service advertisement to its own domain leader (which marks itself as a SL automatically) to announce the content it will distribute. The SL then broadcasts this message to the other leaders. After further forwarding this advertisement to nodes within its domain, each leader waits for the responses from the possible receivers that are interested in the multicast content.

We assume that each DTN node has a unique endpoint ID [4]. Once receiving the *SESSION_ADV* message, an intended receiver is required to send a *SESSION_JOIN* message with its endpoint ID to its own domain leader for the purpose of registration. The leader then puts this receiver into a *receiver list* and marks itself as a RL. It generates a new *SESSION_JOIN* message which only contains the leader's endpoint ID rather than the whole list of all the current receivers and forwards the message in the leader layer. In this way, the membership information in the domain layer is integrated by treating multiple receivers in one domain as a whole receiving endpoint.

Membership integration also happens in the leader layer. During the procedure of the service advertisement, a leader may receive multiple *SESSION_ADV* messages. It accepts the first *P* advertisements and takes the sources of these messages as its parents, where *P* is a configurable protocol parameter that controls the entire redundancy. Once a leader becomes a RL, it then sends a copy of *SESSION_JOIN* message to each parent. The parent that receives this message will mark itself as a FL and records the endpoint ID of the RL into a *forwarder list*. This new FL then forwards a new *SESSION_JOIN* message which only contains its own endpoint ID to each of its parents. In this way, a FL hides the membership information of its downstream leaders to its parentss. Such operation continues and eventually the SL also becomes a FL. Considering *n* receivers distributed in different domains. The SL does not need to keep the membership information of all these receivers. Its forwarder list at most contains *m* FLs, where *m* is the number of its out-degrees in the leader layer and in general is much smaller than *n*.

A *SESSION_LEAVE* message is required to send when an intended receiver wants to deregister its membership, or the receiver list of a RL or the forwarder list of a FL becomes empty (which turns a RL and a FL back to be a NL). When receiving a *SESSION_LEAVE* message, the RL and the FL will remove the source of message from its receiver list and forwarder list respectively. Therefore, such membership variation only affects FLs when necessarily. When the forwarder list of the SL is empty, it turns to be a NL too.

C. *Multicast Structure Construction and Maintainence*

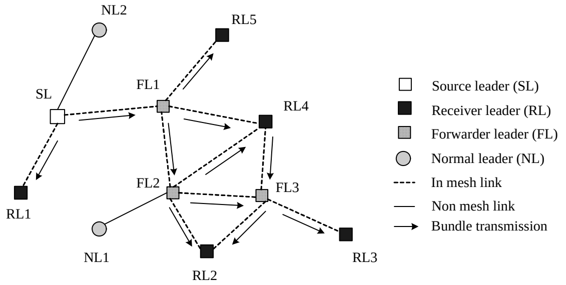
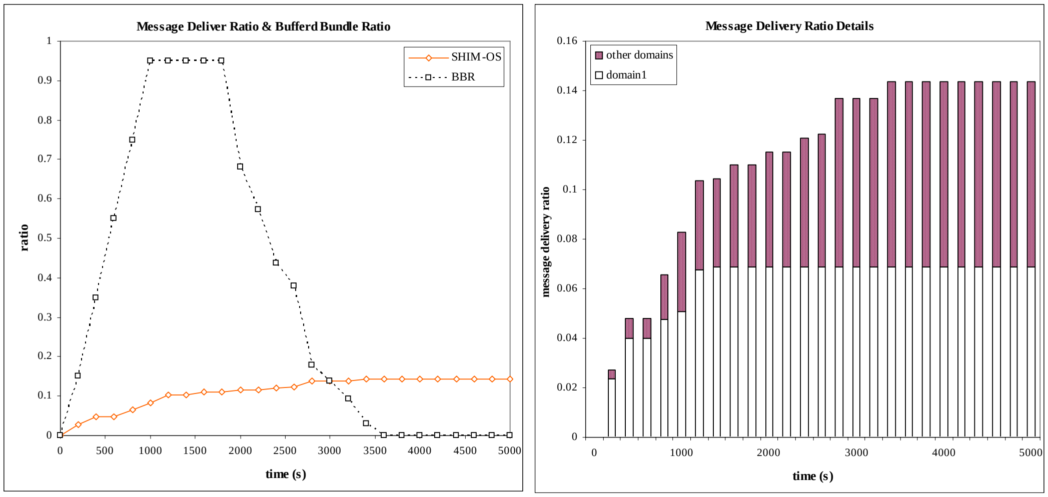


Figure 3. An example of the multicast mesh created by SHIM in the leader layer with P = 2.

When the SL has a multicast message, it delivers the bundle to each



(a) (b)

Figure 4. (a) Message delivery ratio and buffered bundle ratio of SHIM-OS with 1 source and 8 receivers. (b) Message delivery ratios of domain1 (where the source is located) and the sum of that of the other three domains

destination in its forwarder list. Each bundle maintains a pending list L_P. Initially, the SL sets L_P as the same as its forwarder list. It then checks if there is currently an available link to reach a downstream FL in L_P. If a FL is accessible, the SL then forwards a copy of the bundle to this destination by DTN unicast method and removes it from L_P. When L_P becomes empty, the bundle is then considered as being successfully transmitted and the SL starts to process the next bundle. Otherwise, the bundle will be stored in the local buffer of the SL and wait for the next opportunity to be forwarded. Periodically, the SL checks the availability of the outgoing links destined for each destination in L_P. It retransmits the buffered bundle immediately once there is a chance to reach a destination and updates the contents of L_P. Once receiving a bundle, a FL behaves the same as the SL. Bundles are forwarded to the next FL in its forwarder list according to the current accessibility of the associated outgoing links. Such link availability information is retrieved from a knowledge oracle as the design of DTBR or provided by the underlying DTN unicast method as the design of OS-Multicast. Therefore, a multicast mesh is constructed in a hop-by-hop manner in the leader layer. Bundles are then transmitted along this mesh until they reach the RLs and then further forwarded to the intended receivers via the intra-domain DTN multicast method supported in the domain layer. Fig. 3 shows an example of the mesh constructed by SHIM.

The multicast mesh is not required to be connected as the connectivity in the leader layer keeps changing. A branch between a FL and one of its downstream FL in the mesh will be removed when the forwarder list of the downstream FL is empty. And new branches are added automatically into the mesh as new RLs and FLs occur in the leader layer.

III. PERFORMANCE EVALUATION

We implemented two versions of SHIM: SHIM-DTBR and SHIM-OS with different intra-domain method using *ns*-2 simulator (version 2.28). We also extended DTBR and OS-Multicast to conduct the inter-domain multicast operations, called Flat-DTBR and Flat-OS respectively. Basically, Flat-DTBR and Flat-OS treat multiple domains as one big group and create a huge flat tree to deliver bundles to those receivers distributed in different domains. Both require the source maintains a whole list of all the receivers. During the bundle transmissions, Flat-DTBR divideds the receiver list at each branching node in the multicast tree, while Flat-OS requires each intermediate node maintains the whole receiver list as the source. Performance comparisons between DTBR and OS-Multicast are detailed in [15].

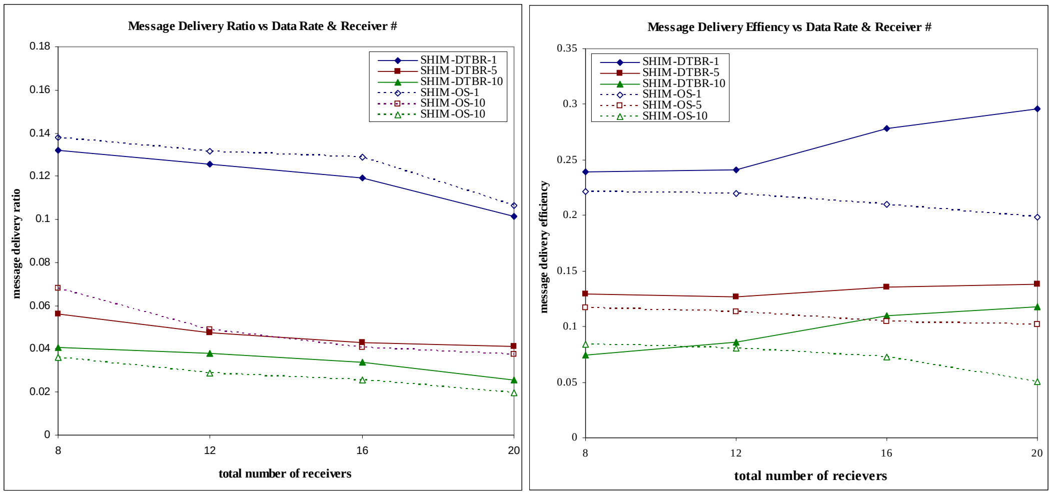
Out performance metrics include: *i*) *message delivery ratio*, which is defined as the number of unique multicast bundles successfully delivered to all receivers over the total number of unique bundles that are expected to be received; *ii*) *message delivery efficiency*, which is the ratio between the unique bundles received by the receivers and the total traffic generated in the network.

We randomly deploy 40 DTN nodes in a 2500m × 2500m area. The simulation is uniformly divided into 4 sub-areas (1500m×1500m) with overlaps. Each sub-area has 10 nodes to form a DTN domain. A domain leader is randomly selected from these 10 nodes. The nodes move in each sub-area following the ZebraNet mobility pattern [16]. To simplify our implementation, we make the leader of domain1 as the multicast source. It generates bundles in the first 1,000 seconds with rate of 1bundle/second to 10 bundles/second and then stops. The number of receivers within each domain varies from 2 to 5. The control parameter *P* is set to be 3.

A. *Prelenimary Results*

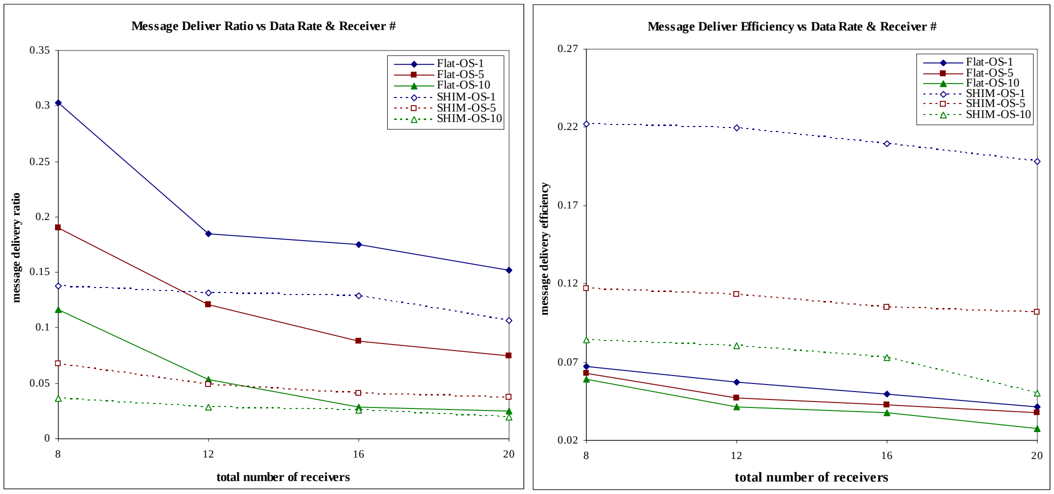
First, we illustrate the effectiveness of SHIM. In this test, we set the simulation length as 5,000 seconds and the source data rate as 1 bundle/second. Each domain has 2 receivers (8 receivers in total). We collect the message deliver ratio of SHIM-OS over the simulation time and calculate the buffered bundle ratio (BBR) which is defined as the ratio of the unique bundles stored in the non-receiver nodes in the networks over the total generated bundles. The simulation results are shown in Fig. 4.

In the first 1,000 seconds, BBR quickly increases as the source keeps generating multicast messages. The message delivery ratio of SHIM-OS also increases in this period of time. It contains two parts: *i*) the successful delivery from the source to the receivers within domain1 by OS-Multicast, denoted as the inside delivery ratio; and *ii*) the successfully delivery from the source to the receivers distributed in the other domains by SHIM operations, denoted as the outside delivery ratio. As shown in Fig. 4 (b), the outside delivery ratio increases more slowly increases than the inside delivery ratio because of the delays of forwarding bundles from the SL to the RLs. After 1,000 seconds, the source stops generating bundles and most data are now stored in the intermediate nodes. Therefore, BBR becomes stable. SHIM-OS still



(a) (b)

Figure 5. (a) Message delivery ratio of SHIM-OS-x compared to that of SHIM-DTBR-x. (b) Message delivery efficiency of SHIM-OS-x compared to that of SHIM-DTBR-x.



(a) (b)

Figure 6. (a) Message delivery ratio of SHIM-OS-x compared to that of Flat-OS-x. (b) Message delivery efficiency of SHIM-OS-x compared to that of Flat-OS-x.

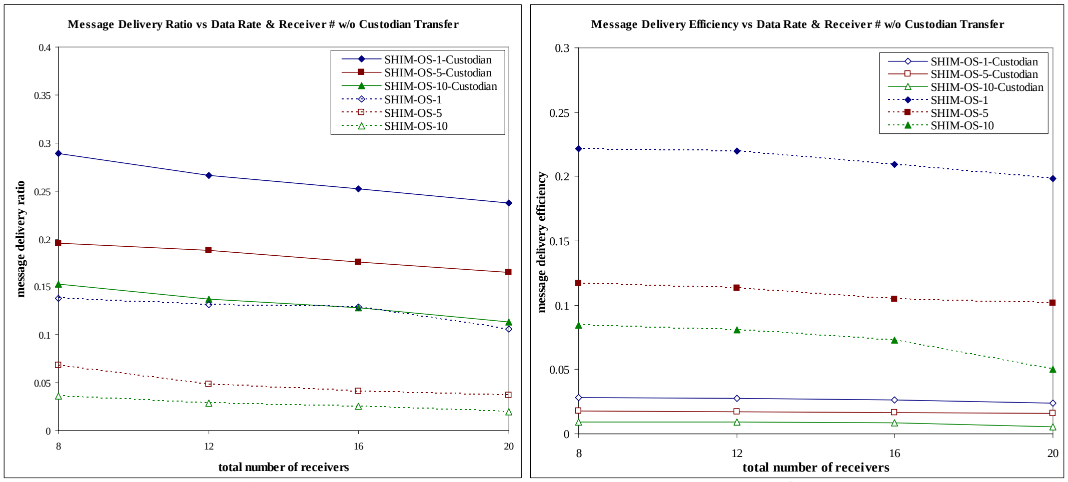
works and the outside delivery ratio keeps increasing. The BBR quickly drops from 1,800 seconds due to two reasons: *i*) buffered bundles are delivered to the intended destinations; and *ii*) bundles are dropped due to the collisions and buffer overflows when a large amount of message retransmissions compete for the limited system resources. After 3,400 seconds all the stored bundles are either transmitted to the receivers or dropped when BBR becomes zero. We can tell that the design of SHIM is effective because the outside delivery ratio stops increasing only when there is no bundles kept in the intermediate nodes. In the rest of our tests, we set the simulation length as 3,000 seconds because the message delivery ratio doesn't improve much after that.

B. *Performance Evaluations*

We then vary the number of receivers in each domain from 2 to 5 and the source data rate from 1 to 10 bundle(s)/second. Fig. 5 shows the comparisons between SHIM-OS and SHIM-DTBR. The term of SHIM-OS-x represents the results of SHIM-OS when the traffic rate is *x* bundle(s)/second.

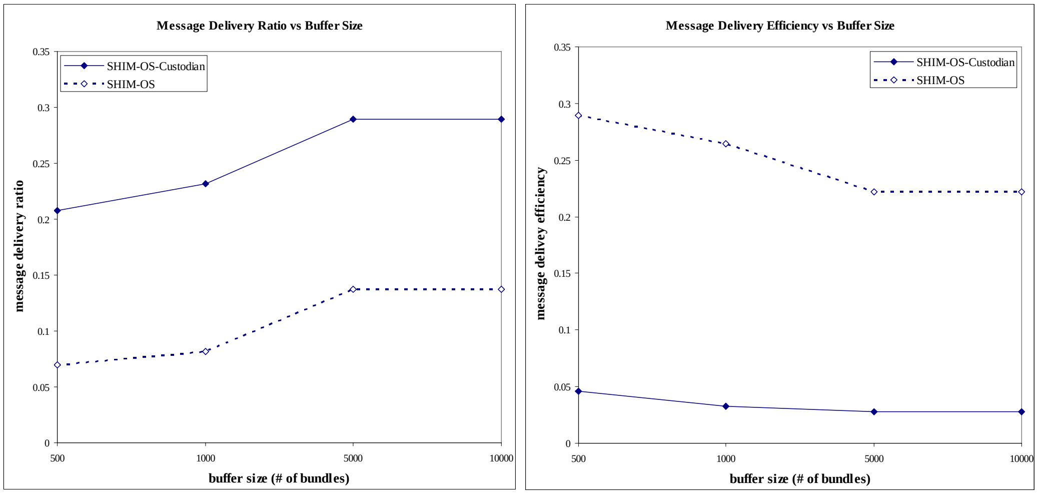
Fig. 5 shows that: *i*) the message delivery ratio of SHIM decreases as the source data rate increases, no matter whether DTBR or OS-Multicast is chosen as the intra-domain multicast scheme. The large amount of bundles generated by the high source traffic rate will trigger a lot of retransmissions in the intermediate nodes. These retransmitted bundles compete with the newly generated bundles for the limited bandwidth. Thus, the message delivery ratio of SHIM drops; *ii*) the message delivery ratio of SHIM-OS is better than SHIM-DTBR when the traffic rate is light but becomes worse when the traffic rate is heavy. Similar observations have been reported and explained in [15]; *iii*) the efficiency of SHIM-DTBR is better than SHIM-OS because OS-Multicast is designed to deliver bundles using multiple paths. Therefore, it improves the message delivery ratio but introduces more redundancy than DTBR.

The comparison results between SHIM-OS and Flat-OS are illustrated in Fig. 6. It shows that: *i*) Flat-OS can deliver more bundles to the receivers than SHIM-OS. In Flat-OS, a huge flat multicast tree that spans receivers in different domains is created and maintained. It treats the domain leaders as the same as the non-leader nodes.



(a) (b)

Figure 8. (a) Message delivery ratio of SHIM-OS compared to that of SHIM-OS-Custodian with varying the number of receivers and source traffic rate. (b) Message delivery efficiency of SHIM-OS compared to that of SHIM-OS-Custodian.



(a) (b)

Figure 9. (a) Message delivery ratio of SHIM-OS compared to that of SHIM-OS-Custodian with varying the local buffer size. (b) Message delivery efficiency of SHIM-OS compared to that of SHIM-OS-Custodian.

Therefore in Flat-OS, bundles are not required to be forwarded to the RLs before they are delivered to the real receivers. However, SHIM-OS requires that the inter-domain traffics must be handled by the domain leaders. Thus, Flat-OS can utilize more opportunistic links in the networks to help deliver bundles; and *ii*) the efficiency of SHIM-OS is much better than that of Flat-OS because of its hierarchical infrastructure. In Flat-OS, every intermediate node has to maintain the multicast states for all the receivers, while SHIM-OS distributes such information to different domains. More redundancies over all the domains are generated in Flat-OS and worsen its efficiency.

C. Performance Enhancement with Custodian Transfer Enabled

To further improve the message delivery ratio, we then integrate the custodian transfer functionality (CTF) into the implementation of SHIM-OS. Basically, the custodianship of a bundle will not be released by its current host until a bundle ACK is received from the next hop. We set up the simulation length as 10,000 seconds to see the long-term improvement caused by CTF. Compare to the results depicted in Fig. 4. (b), the results in Fig. 7 shows that: *i*) the message delivery ratio achieved by SHIM-OS with the custodian transfer function enabled is more than twice of that of SHIM-OS without CTF; *ii*) the overall delivery ratio keeps increasing because the lost bundles in the lower layer can be recovered by CTF if a proper ACK is not received; and *iii*) at time 10,000 seconds, the receivers in domain1 (where the source is located) almost receive all the bundles since the percentage of the messages they received is almost 25% (recall that there are in total 4 domains in our simulation setup) of the total messages. The overall message delivery ratio continues to increase when the simulation length is enlarged.

Fig. 8 shows the performance comparisons between SHIM-OS and SHIM-OS-Custodian, SHIM-OS-Custodian achieves much better message delivery ratio than SHIM-OS due to the recovery of the lost bundles by CTF. However, its efficiency becomes much worse as the result of the large amount of redundant traffic introduced by retransmitting the lost bundles. Then we adjust the local buffer size from holding at most 500 to 10,000 bundles, for 8 receivers and the traffic rate of 1 bundle/second. Fig. 9 shows that the message delivery ratios of both SHIM-OS and SHIM-OS-Custodian benefits from the increase of local storage within each DTN nodes. However, their efficiencies decrease because more retransmissions are incurred when more bundles can be stored.

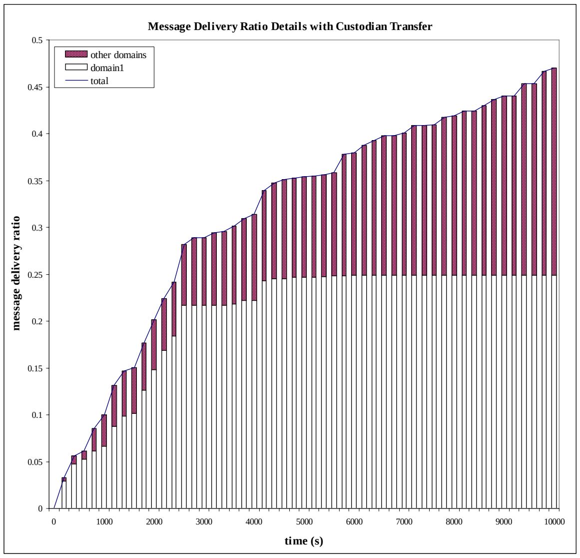


Figure 7. The improvement of message delivery ratio of SHIM-OS by enabling the custerdian transfer function.

IV. CONCLUSIONS

In this paper, we proposed SHIM to provide the inter-domain multicast service in DTNs. It builds up a leader layer consists of all the domain leaders and the sources upon different DTN domains. By integrating the membership information hierarchically, it can greatly suppress the membership information maintained at the source leader. The inter-domain data communication between different domain leaders is handled by DTN unicast methods. And any intra-domain multicast approach selected by the domain administrator can be integrated with SHIM.

Our results show that SHIM can achieve much better efficiency than directly extending existing intra-domain DTN multicast approaches to perform inter-domain multicast operations. Its message delivery ratio can be further improved with enabling the custodian transfer functionality and putting more local buffer within each DTN node.

ACKNOWLEDGMENT

This work has been supported by DARPA under Contract W15P7T-06-C-P430. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the sponsor of this work.

REFERENCES

[1] M. Chuah, L. Cheng, and B.D. Davison, “Enhanced disruption and fault tolerant network architecture for bundle delivery (EDIFY)”, Proceedings of IEEE Globecom, pp. 807-812, November 2005.

[2] K. Fall, “A delay-tolerant network architecture for challenged Internets”, Proceedings of SIGCOMM’03, August 2003.

[3] S. Jain, M. Demmer, R. Patra, K. Fall, “Using Redundancy to Cope with Failures in a Delay Tolerant Network”, Proceedings of SIGCOMM’05, Augest 2005.

[4] S. Farrell, V. Cahill. "Delay and Disruption Tolerant Networking," ISBN 1-59693-063-2, Artech House, 2006.

[5] S. Farrell, V. Cahill, D. Geraghty, I. Humphreys, and P. McDonald, "When TCP Breaks: Delay- and Disruption-Tolerant Networking." IEEE Internet Computing, vol. 10, no. 4, 2006, pp. 72-78.

[6] S. Jain, K. Fall, R. Patra, "Routing in a delay tolerant networking", Proceedings of SIGCOMM’04, Augest 2004.

[7] Z. Zhang, "Routing in Intermittently Connected Mobile Ad Hoc Networks and Delay Tolerant Networks: Overview and Challenges." IEEE Communications Surveys and Tutorials. Jan, 2006.

[8] Q. Ye, L. Cheng, M. Chuah, and B. Davison, “OS-multicast: On-demand Situation-aware Multicasting in Disruption Tolerant Networks”, Proceedings of IEEE 63rd VTC, Vol. 1, pp. 96-100, Melbourne, Australia, May 2006.

[9] W. Zhao, M. Ammar, and E. Zegura, “Multicasting in delay tolerant networks: semantic models and routing algorithms,” Proceeding of Sigcomm’05 Workshop in DTN, August 2005.

[10] T. Bates, R. Chandra, D. Katz, and Y. Rekhter, “Multiprotocol extensions for BGP”, Internet Engineering Task Force (IETF) RFC 2283, Febeurery 1998.

[11] S. Kumar, D. Thaler, C. Alaettinoglu, D. Estrin, and M. Handley, “The masc/bgmp architecture for inter-domain multicast routing”, Proceedings of ACM SIGCOMM ’98, September 1998.

[12] Y. Rekhter, T. Li, “A Border Gateway Protocol 4 (BGP-4)”, IETF RFC-1771, March, 1995.

[13] S.H. Bae, S.-J. Lee, W. Su, and M. Gerla, “The design, implementation, and performance evaluation of the on-demand multicast routing protocol in multihop wireless networks”, *IEEE Network*, pp.70–77, January 2000.

[14] C. Gui and P. Mohapatra, “Overlay Multicast for MANETs using dynamic virtual mesh”, *Wireless Networks*, vol. 13, pp. 77-91, 2007.

[15] Q. Ye, L. Cheng, M. Chuah, and B. Davison, “Performance Comparison of Multicast Approaches in Disruption Tolerant Networks”, technical reports of Lehigh University, LU-CSE-06-020, 2006.

[16] Y. Wang, S. Jain, M. Martonosi, and K. Fall, “Erasure Coding Based Routing for Opportunistic Networks”, Proceeding of ACM Sigcomm Workshop in DTN, August 2005.