Team Oriented Multicast: a Scalable Routing Protocol for Large Mobile Networks

Yunjung Yi, Mario Gerla, Joon-Sang Park, and Dario Maggiorini

University of California at Los Angeles
California, USA
`{yjyi, gerla, jspark, dario}@cs.ucla.edu`

Abstract. [1]
This paper proposes a multicast protocol, called *Team Oriented Multicast* (TOM). TOM builds up a "motion aware" hierarchy to support efficient, scalable team multicast protocol. TOM identifies clusters of nodes with same affinity as teams and manages the multicast membership information using the unit of team rather than dealing with individual node members. TOM uses a two-tier data dissemination approach where the source propagates a data packet to each subscribed teams leader and each leader forwards the data to the entire team. TOM constructs a multicast mesh structure among leaders of subscribed teams, where each leader is connected to m other parent leaders, receiving duplicate packet streams from each parent. Each team leader proactively maintains the list of nodes in the same multicast mesh.

Simulation results show the effectiveness, scalability and reliability of TOM in various representative scenarios.

1 Introduction

With the advances in wireless ad hoc communications, robotics and microflyer technology, the deployment of large-scale networks with hundreds and even thousands of distributed autonomous nodes will be possible in the near future. In such large scale networks, with no fixed infrastructure, providing an efficient, scalable routing and multicast scheme is extremely challenging. In [13], the authors have shown that a hierarchical routing is essential to achieve adequate performance in very large networks. A hierarchical approach, where multicast group receivers are grouped into a few clusters, can be exploited if a stable cluster platform can be maintained. By grouping receivers, QoS protocols consider only a small number of representative nodes instead of thousands of individual members. However, the assumption of a stable cluster platform often fails in MANET scenarios where nodes move quickly and thus the membership of a cluster is extremely fragile. With an unstable cluster structure, hierarchical multicasting may not be a good solution due to excessive cluster maintenance cost.

---

[1] This work is supported in part by ONR "MINUTEMAN" project under contract N00014 - 01 - C - 0016

That observation leads us to conclude that developing a hierarchical multicasting protocol working for all possible scenarios is probably not feasible. Fortunately, in many large scale MANET scenarios (e.g., warfront activities, search and rescue, disaster relief operations, etc.), the mobile nodes are organized in teams with different tasks and, correspondingly, different functional and operational characteristics. In particular, nodes in the same team will have the coordinated motion. We call this model the "affinity team model". For example, various units in a division can be organized into companies and then further partitioned into task forces based on their assignments in the battlefield. In a highway, platoons of cars can be treated as a team because of their motion affinity. Other examples are search and rescue operations, disaster monitoring, and mobile sensor platforms. Our basic observation of those applications is that nodes can be grouped based on their physical location, mobility, or interests. With the affinity team model, it suffices for mobility management to keep track of only one of the nodes in each team (a representative node). Other nodes in the team can be reached through the representative node. As our affinity team model guarantees the stability of clustering (teams) in some degree, the design of an efficient scalable hierarchical multicast structure is now realistic.

Our proposed idea, *Team-Oriented Multicast* (TOM), exploits the affinity team model. It defines teams and manages the membership information using the unit of team rather than that of a set of individual nodes. A team is defined as a set of nodes that have the motion affinity and interests differentiated by subscribed multicast groups. To fully utilize that logical hierarchy of teams, TOM provides a two-tier multicasting approach where the source propagates a data packet to each subscribed team's leader and each leader forwards the data to the entire team. As one can easily expect, the performance of such a two-tier approach considerably depends on the design of first-tier communication platform among leaders. From now on, we will call the leader the "team representative node" (TRN). If the reliability and latency of data transmission to each TRN can be bounded, this two-tier approach can provide a reasonable throughput. Otherwise, this approach may perform worse than a flat multicast protocol such as ODMRP [14], because of the extra overhead to manage the logical cluster architecture. In Internet multicast, shared tree structures are often used to improve the efficiency of multicasting. Internet multicasting protocols emphasize efficiency rather than reliability because the underlying wired medium guarantees the data delivery in some degree. In MANET scenarios, this is not true anymore. Due to collisions, congestion, link errors, jamming, asynchronous links and interferences, the delivery ratio on a wireless connection varies over time and it may becomes unacceptable(e.g., less than 60%) [7]. The delivery ratio of a packet sharply drops as the traveled hops increase [7]. This unique characteristic makes the hierarchical MANET multicasting protocol distinctive from hierarchical multicasting protocols proposed in wired network. Thus, the main focus of TOM is to provide an efficient and robust platform among selected team leaders.

The rest of paper is organized as follows. Section 2 briefly overviews the related works. In Section 3, we will discuss the design issue and protocol description of TOM. Following Section 4 will show the evaluation of TOM through simulation study. Finally, we conclude our paper in Section 5.

2 Related Works

As the node mobility is one of main challenges to design MANET routing protocol, many researches have been conducted to develop a mobility model [3] [6]. The observation of group affinity is not new. In [3] [10] [16], the author already proposed a group mobility model where a set of nodes move together. There are many researches on clustering algorithms and routing algorithms considering node mobility [4] [2] [9] [8]. However, not many researches have been accomplished on hierarchical MANET multicasting protocols working with group mobility.

A few MANET multicasting protocols choose hierarchical approaches [12] [17] [5] [21]. Those ideas have been mostly focused on the efficiency and reliability in a rather small scale network. Unicast tunneling used in AMRoute and unicast transmission in MCEDAR are not scalable, since the cost of unicast grows as the number of participants or *cores* increase.

In [21], the authors proposed a hierarchical multicasting based on the scalable unicast routing LANMAR [8], called M-LANMAR. The approach and design goals of M-LANMAR are similar to TOM. M-LANMAR, however, totally depends on the underlying unicast protocol to propagate the packet to landmarks, and thus it shows the limited scalability.

TOM, divergent from previous approaches, addresses the low packet reception rate in a large network and provides a robust forwarding structure. This is important especially in a large-scale network where the cost (e.g., latency and packet overhead) of packet recovery is considerably high.

3 Algorithm Description

As a first step to a hierarchical multicasting, TOM constructs a virtual hierarchy by organizing nodes to a few teams based on affinity team model and selecting a leader for each team. With such a hierarchy, TOM provides a two-tier multicasting paradigm where the source delivers the packet to each member in two steps: (1) inter-team data forwarding: data forwarding to each team leader called a team representative node (TRN) and (2) intra-team forwarding: data dissemination within a team initiated by the TRN node. Detailed algorithm description is found in our full version paper [20].

The network that TOM considers consists of several teams $\{T\}$ and individual nodes that do not belong to any team due to the lack of affinity. A team T is a connected un-directed graph with the maximum distance D from a node i to j (i and $j \in T$). A link (i, j) implies a direct connection between i and j. A team T is defined as a set of nodes having the same mobility pattern and common interests i.e., motion affinity group. Each node discovers a team and selects a leader in a distributed manner based on the idea proposed in [9]. In the paper, for the sake of simplicity, we assume: (1) a node does not join a multicast group if it does not belong to a team; and (2) all nodes in the same team subscribe the same multicast groups. With those assumptions, inter-team membership maintenance and data forwarding become simple. Thus, this paper focuses on inter-team membership management and data forwarding.

3.1 Inter-team Group Membership Management

TOM builds up a m-ary connected multicast mesh structure among subscribed teams' leaders. In m-ary connected multicast mesh structure, each leader has at most m undirected connections with other leaders. By allowing m redundant packet receptions from connected leaders, note that each node forwards a data packet to all connected leaders except toward incoming direction, our mesh structure provides a reliable transmission platform over a tree structure. To effectively manage the mesh structure with dynamic membership changes, TOM develops a mesh maintenance algorithm, where the goals of algorithm are (1) requiring less dynamic mesh re-construction; (2) working in a distributed fashion; and (3) demanding low overhead.

To maintain a path between two leaders connected in the multicast mesh structure, TOM uses a distance vector routing protocol (DSDV). With random mobility of each team, all nodes should proactively manage the paths to leaders. Thus, each node in the network maintains the table of all the leaders who subscribed to any group and periodically exchanges and updates that table with neighbor nodes. We call the table of leaders as TRN table hereafter.

Our mesh structure is an undirected connected graph $G = (V, E)$. Each vertex $v \in V$ can have at most m edges. The redundancy factor m can be adjusted considering the overall reliability. However, to satisfy the connectivity of our graph model, m should be greater than 2 (two) [20]. As default, we use $m = 3$. Each vertex v in the graph has a unique sequence number seq_v, which is assigned at Membership Join phase. A root vertex $r \in V$, which has the lowest sequence number among V, maintains the vertices list V and the current sequence number $C(seq)_G$ to assign a new member. The sequence number is important to maintain a connected graph with dynamic membership changes i.e., new join, leave or link changes.

A Group Membership Join: TOM, because of TRN table update mechanism, can propagate the partial membership information with low overhead to the entire network. Only root vertices of multicast groups advertise the group address and the size of the multicast mesh graph to the entire network by piggybacking the information on TRN table exchange messages so that a new team can finds a point to send a Join Query by looking up its TRN table.

The Join of a new team T_i to a group m_j is a procedure to add a vertex trn_i (the leader of T_i) and edges with the minimal cost to the multicast mesh graph $G(m_j)$ while keeping $G(m_j)$ connected. When a new team T_i wants to join a multicast group m_j, the leader trn_i of T_i first looks up its local TRN table

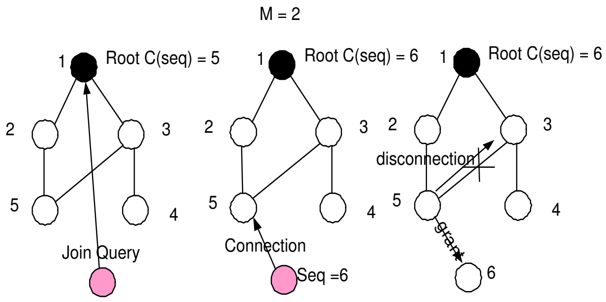


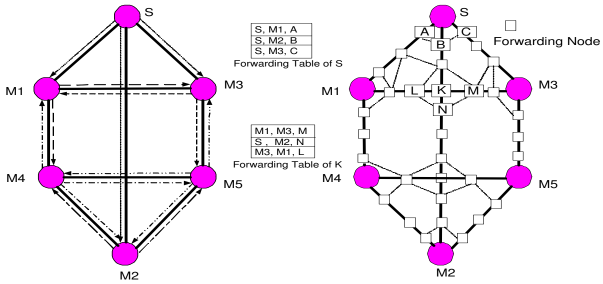
Fig. 1. Join Procedure

to retrieve the root vertex of $G(m_j)$ and sends a query if available. Otherwise, (i.e., this node is a new incoming node or no team has subscribed to m_j), trn_i claims itself as a root vertex in a graph $G = (\{trn_i\}, \emptyset)$ and starts advertising the membership information with TRN table exchange. Once a root vertex discovers another graph for the same group, it tries to merge two graphs (**Graph Merge Procedure**)(see [20]).

When a root vertex r receives the query packet, it increments the current sequence number $C(seq)_G$ and assign to trn_i (i.e., $seq_{trn_i} = C(seq)_G$). r returns the member list V and new sequence number seq_{trn_i} to trn_i. Each node has two connection list: the parents list CL_p and children list CL_c. For each link (v, w) where seq_v < seq_w, v is a parent of w and w is a child of v. To guarantee a connected graph, a vertex v should have at least one link $e_p = (v, w)$ where seq_w < seq_v (i.e., $CL_p \neq \emptyset$) (The proof is given in [20]). trn_i sorts the member list V in ascending order according to the distance from trn_i based on its TRN table. Until, trn_i finds a parent node to connect, it sends a Connection Request to a node v_j i.e., j-th element in V. Upon receiving a Connection Request packet, v_j performs **Connection Establish Procedure** (see [20]). Without a link and node failure, trn_i will find at least one parent node if $m \geq 2$ (see [20]). Note that we assume that network is not partitioned.

Once trn_i is connected to G, then trn_i informs the root vertex r. The root vertex adds trn_i to V and propagates to G with the current sequence number $C(seq)_G$. To provide resilient membership maintenance in spite of a failure of the root, we duplicate the membership information to each vertex in the graph. Fig. 1 illustrates an example of Join Procedure. Once a node joins a group, it may add connections up to m links adaptively.

Membership Leave When a team leaves a group, the leader sends an explicit Membership Leave Request. If the leader is not a root vertex, it disconnects all connections and informs the root vertex. If a root vertex wants to leave, it chooses the vertex v with the smallest sequence number and hands over the root role. A new root advertises the entire nodes in the graph the change of root address. A root without edge simply stops advertising so that each node in the network removes the entry from TRN table after a timeout.



(a) A Multicast Mesh(G1)

(b) A Virtual Forwarding Structure

Fig. 2. Multicast Mesh and Virtual Forwarding Structure

3.2 Inter-team Data Forwarding

Inter-team data forwarding mechanism is the key to the success of TOM's two-tier data transmission approach. To design an efficient and reliable inter-team forwarding mechanism, however, is very challenging, since the average distance of data transmission is very large and the path reliability is extremely low. The path redundancy by the mesh structure does not significantly improve the reliability without a bounded packet reception rate between two connected leaders. Thus, our main goal of inter-team data forwarding scheme is to improve the path reliability in an efficient way.

TOM proposes the multi-path neighbor aggregation (MPNA) technique. MPNA builds a virtual forwarding structure including intermediate nodes and leaders in the multicast mesh. Fig. 2 illustrates an example of a multicast mesh structure and following a virtual forwarding structure. Ideally, a node in the virtual forwarding structure should relay a packet only once for efficiency. In MPNA scheme, however, a node may relay the packet more than once to propagate aggregation information, if necessary. We should note that this forwarding node concept is not new. It was already proposed in ODMRP. However, MPNA develops a different mechanism to find forwarding nodes. In ODMRP, each intermediate node sets the forwarding flag, if it receives a Join Reply packet from a neighbor, thus explicit control messages are necessary. In MPNA, a sender calculates next forwarding nodes (next hops) using TRN table and adds the aggregation header piggybacking the information to the packet. A node sets the forwarding flag, if it discovers that, by examining the aggregation header of incoming packet, a previous hop selects it as a next hop. Conceptually, it is more similar to soft-state Differential Destination Multicast (DDM) [11] than to ODMRP. In DDM, targeting a small group, each source aggregates the packet in a similar way to MPNA. DDM, however, attempts to reduce the aggregation information by deploying the synchronization between a node and the next hop. If a route is pretty stable, DDM can significantly reduce the aggregation overhead. TOM, however, is designed for a network with high mobility, and thus, the stability of a path is pretty low. More importantly, the underlying routing protocol used to update paths between leaders, DSDV, tends to change routes frequently. With DSDV, a node updates a path whenever it discovers a fresher route even though the current path is still valid [7]. Thus, the optimization of DDM is not directly applicable to TOM. Furthermore, TOM provides multi-path transmission. Thus, it differs from previous schemes [14, 11].

3.3 Intra-Team Membership Maintenance and Data Forwarding

We use a simple approach to handle intra-team membership. This is warranted by the fact that within the team, relative mobility is minimal and only short range because of team affinity. To maintain the team, e.g., the leader re-selection, team forming and team split/merge, each node is required to periodically exchange some information. In our implementation, each node exchanges local routing table including entries in $\frac{D}{2}$ hops from a node and a leader is selected based on the routing table information. Without deploying explicit membership join/leave messages, nodes can advertise the membership by piggybacking on the routing table update packets. The data is propagated within a team using a “scoped flooding”.

4 Simulation Study

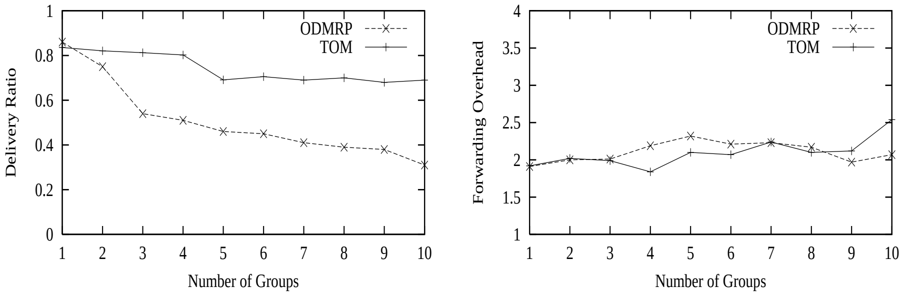
In this section, we evaluate the performance of TOM through extensive simulation experiments. As a reference for performance comparison we use ODMRP (On-Demand Multicast Routing Protocol) [14]. This benchmark choice is justified by the fact that ODMRP was shown to outperform most of the existing ad hoc multicast schemes such as CAMP [12], AMRoute [5] and ARMIS [19] in mobile scenarios [15].

Our performance metrics are as follows: (1)delivery ratio: The ratio of the number of delivered packets to each member versus the number of supposedly received packets by each member; (2) forwarding overhead: the total number of forwarded data packets versus the total number of delivered packets to members; and (3) packet latency: the average end-to-end delay of a multicast packet to each member.

We use QualNet [1] simulator, a successor of GloMoSim [18]. It provides a detailed and accurate model of the MAC, Channel and routing protocols. We use default parameters provided by QualNet. In our simulation, each source generates data in a CBR (Constant Bit Rate) fashion with UDP (User Datagram Protocol). Each source generates 4 pkts/second with 512 bytes packet size. We use IEEE 802.11 DCF MAC and two-ray ground path-loss model for the Channel. The transmission range of each node is 376m and bandwidth of the device is 2Mbits/sec.

In the network, 1000 nodes are uniformly placed within 6000 x 6000 m^2 terrain. We divide the network into 36 groups where each group has the same group mobility following “Reference Point Group Mobility (RPGM)” model [3]. Except for the mobility study, for all simulations, each team moves with 10 m/s speed with 10 seconds pause time. We assume that the whole group joins a multicast group if a node in the group joins i.e., a group defines a team if it subscribes a multicast group. Thus, maximally 36 teams can exist in the network. The average number of neighbors for each node is 10 and the scope of a team is four. For maintaining the routing structures, ODMRP uses 2 seconds interval for each Join Query and TOM uses 1 second interval for TRN table update. To maintain a team i.e., for a cluster management, each node periodically broadcast its local routing table at every 5 seconds. In our simulation study, we omit the team discovery procedure. We assume that a team is pre-fixed for the simplicity of the evaluation.

TOM, as default, uses a multicast mesh structure with $m = 3$ and MPNA scheme with the path redundancy factor $r = 2$ and a new path update interval $I_{update} = 0.25$ seconds.



(a) Delivery Ratio (b) Forwarding Overhead

Fig. 3. Scalability Test v.s. Group Number

4.1 Study on Scalability

One of our main contributions of TOM is the scalability as the group size and number, and network size increases. To show the advantage of TOM compared to traditional *flat* multicast protocols, we examine the throughput changes of TOM over different group number and size compared to those of ODMRP, a representative *flat* MANET multicast protocol. By deploying a large number of nodes (we use 1000 nodes through our simulation), we implicitly show the scalability of TOM with the large number of nodes. To test the scalability with the group number, we increase the number of multicast group(s) from 1 to 10 where each group has five subscribed teams with a single source. For a group size test, we fix the group number and the source number to 1 and increase the number of subscribed teams from 1 to 10.

Fig. 3(a) and 3(b) show the delivery ratio and forwarding overhead of TOM compared to those of ODMRP with variable group sizes. The forwarding overhead of both TOM and ODMRP slightly grows as the group number increases;

because the network becomes more congested and thus, the delivery ratio degrades. Notably, the delivery ratio of TOM is fairly stable in spite of the increase of offered load. Since TOM does not introduce major control overhead as the group size or number increases, it keeps the network status pretty stable. On the other hand, the performance of ODMRP significantly degrades as the group number increases. As ODMRP applies separate Join Query flooding for each group, the control overhead of ODMRP proportionally increases to the number of group. Thus, ODMRP suffers from heavier load due to the increase of data packets as well as Join Query flood packets as the group number increases. Those results clearly demonstrate the scalability of TOM as the group number increases.

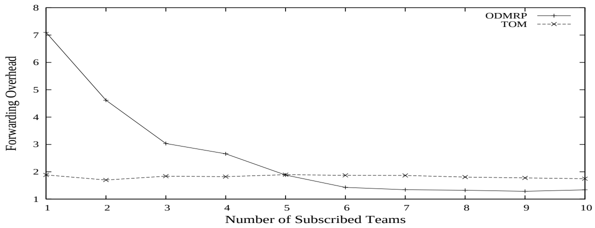


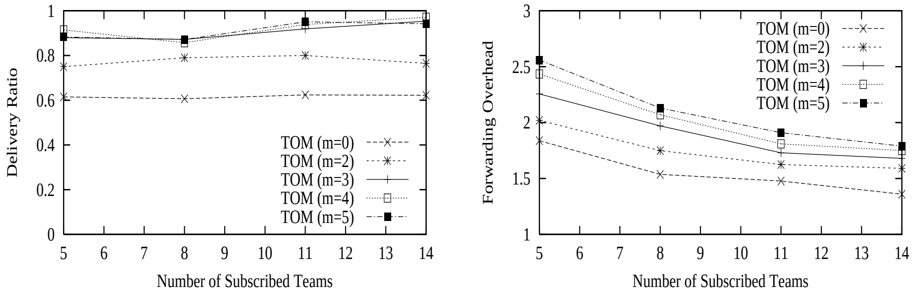
Fig. 4. Forwarding v.s. Group Size

Fig. 4 illustrates the forwarding overhead of both schemes versus the group size. Remarkably, in spite of intra-team flooding overhead, the overhead of TOM is comparable to that of ODMRP. More importantly, the overhead of TOM keeps stable. Note that, if we apply an efficient flooding scheme or ODMRP for the intra-team data forwarding as mentioned earlier, the overhead of TOM can be further reduced. On the other hand, the forwarding overhead of ODMRP is closely related to the group size and actually grows as the group size becomes smaller. Since ODMRP periodically floods a data packet with Join Query message i.e., ODMRP piggybacks the Join Query information on the data packet periodically to update the membership information, the total number of forwarded data packets is dominated by the flooding packets. Thus, the forwarding overhead decreases as the number of members delivering the packet increases. Note that we omit the comparison in terms of delivery ratio with the group size since the next simulation study implicitly shows the result.

4.2 Impact of Mesh Degree on Performance

Intuitively, the packet delivery ratio of a mesh structure will be enhanced as m increases unless the network is congested. In this simulation, we want to investigate the impact of redundancy degree m on the delivery ratio and forwarding overhead. As a reference, we build a 1-level multicast tree with $m = 0$.

For the simulation, we use a multicast group with a single source. We increase the number of subscribed teams from 5 to 14.

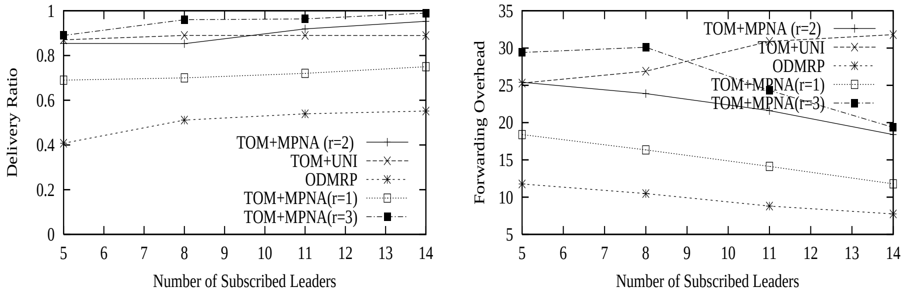


(a) Delivery Ratio (b) Forwarding Overhead

Fig. 5. Mesh Degree Test

In the results, Fig. 5(a) and 5(b), we can observe two major performance improvements between $m = 0$ and $m = 2$ and between $m = 2$ and $m = 3$. The results clearly demonstrate the benefit from the path redundancy created by using the mesh structure. However, a mesh structure with a large redundancy factor more than three does not significantly improve the throughput. Notably, with a mesh with m =5 suffers and performs actually worse than m =4 case due to too heavy forwarding overhead. Empirically, we recommend m =3 to maximize the throughput of the proposed mesh structure.

Note that [20] includes more simulation study results.



(a) Delivery Ratio (b) Forwarding Overhead

Fig. 6. Path Redundancy Test

4.3 Investigation on Forwarding Mechanisms

In this simulation, we investigate the performance of inter-team forwarding mechanisms: (1) separate unicast tunneling; and (2) multi-path neighbor aggregation technique with $r = 1$ i.e., a single path, $r = 2$ and $r = 3$. As we study the throughput of first-tier nodes, we omit the intra-team forwarding in this experiment. Thus, only team leaders become member nodes of a multicast group. And we use fixed team leaders randomly chosen at the initialization of each simulation run. We examine the performance of ODMRP over subscribed leaders, as a reference.

We use a multicast group forming the multicast mesh ($m = 3$) with a single source and variable number of members from 5 to 14.

In Fig. 6(a) and 6(b), we can observe four important facts. First, the delivery ratio of the single-path broadcast schemes used by ODMRP and MPNA with r =1 are remarkably low compared to that of unicast tunneling. Still, the redundant packet transmission in the multicast mesh significantly improves the reliability i.e., TOM+MPNA (r=1) performs far better than ODMRP. Secondly, the multi-path mechanism considerably enhances the throughput. By adding one more path i.e., r =2, the performance of TOM is improved more than 20%. As the throughput difference between r=2 and r=3 is not significant, we recommend to use $r = 2$ for MPNA technique. Thirdly, the forwarding overhead of unicast increases as the number of connections increases. On the other hand, broadcast mechanisms reduce the overhead and efficiently forward a packet by eliminating unnecessary re-braodcasts of the same packet. Thus, broadcast mechanisms are much more scalable than unicast tunneling with group size. Lastly, our virtual forwarding structure becomes more robust and efficient with the group size. The forwarding overhead of MPNA scheme degrades but the reliability of it increases as the group size grows.

Note that, to collect the forwarding overhead of ODMRP in this simulation study, we omit the number of periodic data flooding packets (i.e., Join Query flooding packets). Thus, we consider the overhead of ODMRP as the lower bound to propagate a data within a multicast mesh structure. Considering that each team has many members, TOM does not significantly increase the forwarding overhead even though it applies multiple paths and redundant transmissions.

5 Conclusion

In the paper, we proposed a two-tier hierarchical multicasting protocol exploiting the affinity team model. Our proposed idea, TOM, contributed as follows: (1) by reducing the number of visible members from outside, TOM considerably reduces the complexity and overhead of a multicasting protocol; (2) TOM identified and corrected the low packet delivery ratio in the large-scale network, which should be addressed to develop a scalable MANET protocol; (3) TOM developed a multicast mesh structure and multi-path neighbor aggregation technique to improve the reliability; (4) through extensive study, TOM showed the scalability, reliability, flexibility and efficiency.

References

1. Scalable networks, http://www.scalble-solutions.com.

2. B. An and S. Papavassillou. A mobility-based clustering approach to support mobility management and multicast routing in mobile ad-hoc wireless networks. *International Journal of Network Management*, 2001.
3. G. Bianchi, X. Hong, M. Gerla, G. Pei, and C.-C. Chiang. A group mobility model for ad hoc wireless networks. *Proceedings of ACM/IEEE MSWiM'99*, 1999.
4. B.McDonald and F.Znati. A mobility-based framework for adaptive clustering in wireless ad hoc networks. *IEEE Journal on Selected Areas in Communications*, vol. 17, Aug. 1999.
5. E. Bommaiah, M. Liu, A. McAuley, and R. Talpade. AMRoute: Ad-hoc Multicast Routing protocol. *Internet-draft, draft-talpade-manet-amroute-00.txt*, 1998.
6. T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking*, 2002.
7. D. S. J. DeCouto, D. Aguayo, B. A. Chambers, and R. Morris. Effects of loss rate on ad hoc wireless routing. *technical report MIT-LCS-TR-836*, March 2002.
8. M. Gerla, X. Hong, and G. Pei. LANMAR: Landmark routing for large scale wireless ad hoc networks with group mobility. *Proceedings of IEEE/ACM MobiHOC*, 2000.
9. X. Hong and M. Gerla. Dynamic group discovery and routing in ad hoc networks. *Proceedings of the First Annual Mediterranean Ad Hoc Networking Workshop*, 2002.
10. R. Hutchins and E. Zegura. Measurement from a campus wireless network. *ICC*, 2002.
11. L. Ji and M. S. Corson. Differential destination multicast-a manet multicast routing protocol for small groups. *INFOCOM*, 2001.
12. J.J.Garcia-Luna-Aceves and E. L. Madruga. A multicast routing protocol for ad-hoc networks. *IEEE INFOCOM*, 1999.
13. F. Kamoun and L. Kleinrock. Hierarchical routing for large networks: performance evaluation and optimization. *Computer Networks*, 1977.
14. S.-J. Lee, M. Gerla, and C.-C. Chiang. On-demand multicast routing protocol. *Proceedings of IEEE WCNC*, 1999.
15. S.-J. Lee, W. Su, J. Hsu, M. Gerla, and R. Bagrodia. A performance comparison study of ad hoc wireless multicast protocols. *IEEE INFOCOM*, 2000.
16. M. Sanchez and P. Manzoni. A java based simulator for ad-hoc networks. *http://www.disca.upv.es/misan/mobmodel.htm*, 2001.
17. P. Sinha, R. Sivakumar, and V. Bharghavan. MCEDAR: Multicast core-extraction distributed ad hoc routing. *IEEE WCNC*, 1999.
18. UCLA. Glomosim: A scalable simulation environment for wireless and wired network systems.
19. C.-W. Wu and Y. Tay. A multicast protocol for ad-hoc wireless networks. *MILCOM*, 1999.
20. Y. Yi, M. Gerla, and J.-S. Park. Team oriented multicast: a scalable routing protocol for large mobile networks. *UCLA Technical Report 030034*, 2003.
21. Y. Yi, X. Hong, and M. Gerla. Scalable team multicast in wireless ad hoc networks exploiting coordinated motion. *NGC*, 2002.