Scalable Team Multicast in Wireless Ad hoc Networks Exploiting Coordinated Motion*

Yunjung Yi and Mario Gerla
Computer Science
University of California, Los Angeles
Los Angeles California, USA 90095

{yjyi,gerla}@cs.ucla.edu

Katia Obraczka
Computer Engineering Department
University of California , Santa Cruz

katia@cse.ucsc.edu

ABSTRACT

In this paper, we study a new multicast paradigm for large scale mobile ad hoc networks, namely team multicast. In team multicast the multicast group does not consist of individuals, rather, of member teams. For example a team may be a special task force that is part of a search and rescue operation. The message must be broadcast to each member of each team in the multicast group. Team multicast is very common in ad hoc networks set up to accomplish some collective tasks, such as for emergency recovery or battlefield applications. A key problem in several of the above applications is scalability to large membership size as well as network size. Our approach exploits motion affinity (more precisely, team members' coordinated motion) which is typically present when the set of nodes has a commonality of interests. Each team can be viewed as a logical subnet. Within the team a landmark node is dynamically elected. The addresses of and the paths to the chosen landmarks are propagated into the whole network so that a source of a multicast group can route to the landmark of a subscribed team.

Our protocol, Multicast-enabled Landmark Ad Hoc Routing (denoted as M-LANMAR), uses tunneling from multicast sources to each landmark of the subscribed team and restricted flooding within the motion group. Simulation study shows that M-LANMAR provides efficient and reliable multicast compared with the application of a "flat" multicast scheme (e.g., ODMRP) that does not exploit team coordinated motion.

This paper contains three contributions: a new model for team multicast, with the definition of team dynamics (join, merge, split); the exploitation of team mobility and of landmarks in order to achieve scalable multicast, and; the im-

plementation and performance evaluation of M-LANMAR, a landmark based team multicast scheme.

Keywords

team multicast, MANET multicast protocol, LANMAR, scalability, team-oriented, multicast, mobile ad hoc networks

1. INTRODUCTION

Mobile ad hoc networks (MANETs) are self-organizing networks that do not need a wired/wireless infrastructure. Two nodes communicate directly if they are in the transmission range of each other. Otherwise, they reach via a multi-hop route. Each MANET node must therefore be able to function as a router, forwarding data packets on behalf of other nodes. Because of their unique benefits and versatilities, MANETs have a wide range of applications such as collaborative, distributed mobile computing (e.g., sensors, conferences), disaster relief (e.g., flood, earthquake), war front activities, and communication between automobiles on highways.

In many MANET scenarios (e.g., warfront activities, search and rescue, disaster relief operations, etc.), the mobile nodes are often organized in teams with different tasks and, correspondingly, different functional and operational characteristics. In particular, nodes in the same team will have coordinated motion. We call this model the "affinity team model". For example, attendees of a major conference can be subdivided into teams based on their topic interests for the purpose of organizing birds of a feather sessions; various units in a division can be organized into companies and then further partitioned into task forces based on their assignments in the battlefield.

One of the main challenges of MANET protocol design is the fact that unlike in Internet nodes are moving continuously. In particular, it is difficult to keep track of individual node movements and to route packets to them especially when the network grows large. The "affinity team" model, considerably simplifies the mobility management problem and allows us to design a routing protocol that scales. In fact, it suffices for a source to know the path to one of nodes in the team (say, a landmark) in order to route a packet to any other destination within that team.

Scalable routing in a team situation with coordinated motion has already been addressed in [9] [19]. Here we extend

that concept to team multicast. First, we will make some assumptions about team multicast. Nodes in the same team share the same interest, and thus all the members in the team participate in the same multicast group(s). For example, all the members of a search and rescue task force receive situation updates from other teams. Moreover, multicast dynamics are on a per team basis - and entire team joins or withdraws from a multicast group. Two or more teams may merge into one; or a team may split in sub-teams, depending on the operational needs.

Since MANETs function under severe constraints such as limited bandwidth and energy, group communications should be performed efficient and at low control overhead cost. Several MANET multicast protocols already exist (e.g., ODMRP [16], MAODV [20], CAMP [15], MCEDAR [24]). Anyone of these protocols could be used in a brute force, "flat" approach by treating each node in the team as an individual unit without exploiting the group mobility feature. However, these schemes require periodic or event-driven control packet updates for each member in the multicast group in order to maintain the multicast structure (e.g., membership information, routes, etc.). Those protocols work effectively with small-scale multicast groups (e.g., less than 100 nodes). However, they suffer from severe communications overhead caused by control packet floods (e.g., Join Query or Request packet flooding in ODMRP and MAODV) in a large-scale network with a large number of multicast groups. Such overhead would be unsustainable in a battlefield scenario with multicast groups consisting of dozen of teams, where each team includes hundreds of units.

In this paper, we propose a new approach to multicast in large-scale mobile ad hoc networks. Our approach exploits team motion affinity. For simplicity, we assume that the teams have been predefined and do not change (e.g., merge/split) during the experiment. Distributed algorithms for dynamic recognition of motion affinity among nodes and of formation of groups based on such affinity have been described elsewhere and could be applied here as well [11]. Our multicast protocol works with dynamic team creation. The foundation of our team multicast scheme is Landmark Ad hoc Routing (LANMAR) protocol [9]. LANMAR provides an efficient proactive routing platform which efficiently exploits team mobility. LANMAR assigns a unique "subnet address" to each team. Each team elects a landmark node. The landmark propagates the team address along with the multicast group address(es) the team belongs to into the network using a proactive routing algorithm such as DSDV (Destination Sequenced Distance Vector). As a result, each node in the network has an up-to-date routing entry to the landmark of each team. It also knows which multicast groups are active and which teams belong to each group.

Building on LANMAR, we propose M-LANMAR, a *Multicast-enabled LANMAR* protocol. Our approach works as follows. With the landmark routing information, a multicast source sends multiple copies of the packet to the landmarks in the multicast group. Each landmark then forwards the multicast packet to its associated team via flooding. M-LANMAR has several attractive properties: (1) it dramatically improves the scalability of multicast protocols in team environments. In fact it does not require a separate group

maintenance protocol and extra control packets to exchange or establish multicast membership information. Instead, it piggybacks multicast group information on landmark routing broadcast packets; (2) it improves the reliability and congestion control properties of the multicast protocol. In fact, TCP-like protocols (instead of the traditional UDP) can be used to reliably deliver packets from source to the Landmarks on the unicast tunnels. TCP-like window control will provide the necessary congestion protection. If a real time stream, for example video (instead of data) is being delivered to the teams, an adaptive streaming protocol (friendly to TCP) can be used on the tunnel. Within each team, scoped flooding achieves very good delivery ratio without excessive overhead. Another requirement that can be easier satisfied by the unicast tunnel (than by multicast across landmarks, say) is security.

In summary, the main contributions of this paper are: (1) the identification of team multicast as an important new model in MANETs; (2) the exploitation of mobile "affinity" to render the problem scalable, and; (3) the implementation and evaluation of M-LANMAR, a version of scalable team multicast.

The organization of the rest of the paper is as follows. In Chapter 2, we briefly introduce related work. Our approach, M-LANMAR will be described in detail in Chapter 3 and the simulation results will follow in Chapter 4. After that, we discuss our future works in Chapter 5 and we finally conclude the paper in Chapter 6.

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 [5, 8]. The observation of group affinity is not new. In [5, 14, 22], 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 [6, 3, 12, 9]. However, not many researches have been accomplished on hierarchical MANET multicasting protocols working with group mobility.

In the wired network, a book of hierarchical multicasting protocols have been proposed for the purpose of scalability [4, 23]. As mentioned earlier, the hierarchical multicast protocols in wired network favors the efficiency of multicasting. As our targeting scenarios have different characteristics, those protocols can not be directly applicable to MANET environment.

2.1 MANET Multicast Protocols

Due to its unique characteristics of the MANET such as node mobility, limited resources and very unreliable channel, typical Internet multicast protocols do not perform well in MANET scenarios. MANET multicast protocols should efficiently cope with dynamic topology changes such as fragile multicast tree structure. In general, MANET multicast protocols can be categorized into a tree-based protocol and a mesh-based protocol based on the constructed multicast structure. A tree-based protocol, e.g., MAODV (Multicast Ad hoc On demand Distance Vector), AMRoute (Ad hoc Multicast Routing) [7] and ARMIS (Ad hoc Multicast

Routing protocol utilizing Increasing idnumberS)[26], explores and enhances a multicast tree structure specialized in MANET scenarios. On the other hand, a mesh-based protocol such as ODMRP (On Demand Multicast Routing Protocol) [16], MCEDAR (Multicast Core-Extraction Distributed Ad hoc Routing) [24] and CAMP (Core-Assisted Multicast Protocol) [15] uses a multicast mesh structure that allows redundant paths between a source and a member. With a multicast mesh, members are permitted to receive multicast packets from any their forwarding neighbors, instead of from the only one node (parent) in a multicast tree. Thus, a mesh topology improves the connectivity of a multicast structure and the availability of multicast routes in the presence of dynamic topology changes.

MAODV builds up a multicast tree. In MAODV, a new member that is not a part of the multicast tree floods Join Request packet to the network. Any subscribed member (knows the path to the source) sends Join Reply with the reverse path from the new member. Among multiple received Join Replies, the new member chooses the suitable route (usually the shortest path) and sends an activation message to the replier. A node that has received an activation message is in charge of forwarding packet to the new member. The forwarding node (parent node) uses its routing table to send data to the new member. Upon the link failure, each node maintains the route using on-demand route query floods similar to AODV. MAODV maintains multicast tree structure using prune messages. Notably, MAODV tends to increase control overhead as the offered load increases due to route query flooding packets to maintain routes.

Moreover, AMRoute uses underlying unicast routing protocol to build up a multicast tree. Each group has at least one logical core that maintains the multicast structure. Cores periodically send Join Requests and members send Join Reply. And, ARMIS, without depending on underlying unicast routing protocol, maintains a tree structure. A new member can join the multicast group by sending a unicast Join Query packet to the potential parent. If this join fails, then the new member incrementally broadcasts Join Request until succeeding in joining.

In ODMRP, a mesh-based protocol, group membership and multicast routes are established and updated by the source on demand. When multicast sources have data to send without the knowledge of membership, they flood the Join Query packet to the whole network. A member, upon reception of the Join Query packet, replies "Join Reply" using the reverse path from the source. ODMRP employs a soft-state maintenance of a multicast structure. One member can leave the multicast group simply by not sending the Join Reply back to the sources and thus the sources remove the membership after the timeout. A node, which does not belong to the multicast group but receives the Join Reply packet from the neighbor node, becomes a forwarding node and relays (broadcast) multicast data from the source. To update and maintain the membership information, sources periodically flood the Join Query packets. Each multicast group differentiated by a multicast address maintains the multicast structure separately. Thus, one can easily see that ODMRP is not scalable as the network size grows or the number of multicast groups increases due to the proportional increase of the maintenance cost.

Another mesh-based multicast protocol, CAMP extends the basic approach of the core-based tree (CBT) protocol for the creation of multicast structure with an allowance of one or multiple cores. "Cores" can limits control overhead for members to join the multicast groups. CAMP is based on underlying unicast routing protocol (bellmand-ford routing scheme) to get the correct distance between nodes. A new member that is not part of multicast mesh first sends a Join Request to the nearest core if none of its neighbors are joined to the targeting group. Otherwise, this node advertises membership to neighbor nodes. Without the reachable core, a new member broadcasts Join Request using incremental flooding (i.e., increase the initial TTL upon retransmission of Join Request packet). Any member node can send Join Reply to the new requested member, and the new member chooses the shortest path among multiple replies. With deploying one or multiple Cores for each multicast mesh, CAMP can reduce flooding overhead for Join Request packets. However, CAMP has a limitation of scalability due to the underlying proactive unicast routing protocol. In large scale network, a "flat" proactive routing scheme such as bellman-ford scheme results the huge memory requirement and heavy routing overhead [13].

MCEDAR [24] constructs the set of *cores*, which is a set of dominant nodes that covers all the members in a multicast group. By constructing a multicast mesh among only *cores* and allowing cores to re-broadcast the data packet, where a core covers a set of members, MCEDAR reduces membership maintenance cost and forwarding overhead.

Our proposed idea, however, is divergent from existing MANET multicast protocols in that M-LANMAR aggregates unicast routing table updates and multicast routing maintenance. Thus, M-LANMAR achieves constantly low maintenance cost, because of the underlying hierarchical routing LANMAR, regardless of dynamic membership changes (e.g., the increasing number of members and multicast groups). Furthermore, none MANET multicast protocol, to best our knowledge, effectively utilize affinity team model. We need a new multicast protocol to maximally exploit the group affinity model. LANMAR protocol works effectively with affinity team model [9]. Naturally, we build our multicast protocol by extending LANMAR protocol to support multicast in our targeting scenarios.

3. PROTOCOL DESCRIPTION
3.1 Backgrounds
As a first step to a hierarchical multicasting, M-LANMAR 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, M-LANMAR 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 *landmark* and (2) intra-team forwarding: data dissemination within a team initiated by the landmark node.

The overview of our target system is shown in Fig. 1. The network consists of several teams $\{T\}$ and individual nodes that do not belong to any team due to the lack of affinity. A
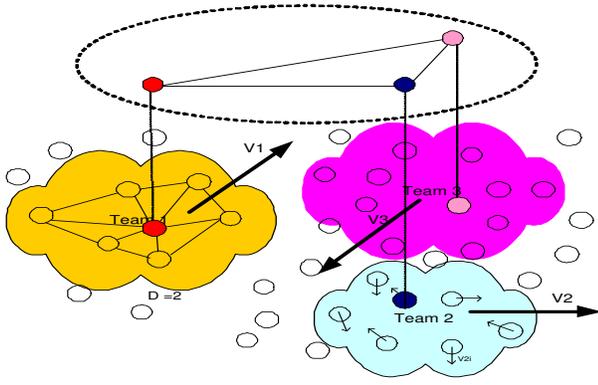
Figure 1: Target System Overview of M-LANMAR protocol

Figure 2: LANMAR Routing Protocol

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 satisfying following conditions:

- same mobility pattern: Our affinity team model assumes that the mobility of a team can be specified as a vector with velocity and direction (see Fig. 1, each team has a mobility Vi). Each node in a team can randomly move within a bounded area ($d * TX$, where $d = \frac{D}{2}$ and TX is the transmission range). Thus, a set of one-hop neighbor nodes of each node may be dynamic, but a set of D-hop neighbor nodes of each node in the same team is stable.

- common multicast group membership: Any two nodes in the same team should have subscribed at least one common multicast group. By subscribing a common group, nodes in the same team share the interest.

- D-hop reachability: Any two nodes in the team are mutually reachable within D-hop distance.

Each node discovers a team and selects a leader in a distributed manner similar to a clustering mechanism [6, 12]. The detail description of full team discovery protocol is out of scope of the paper. Current team management algorithm is based on the idea proposed in [12].

In the paper, for the sake of simplicity, we assume followings:

- a node does not join a multicast group if it does not belong to a team

- all nodes in the same team subscribe the same multicast groups

We address the extension of our protocol without upper assumptions in Section 5.

3.2 Overview of LANMAR Protocol
LANMAR (Landmark Ad Hoc Routing) protocol is a proactive routing [9]. It uses the notion of *landmarks* to keep
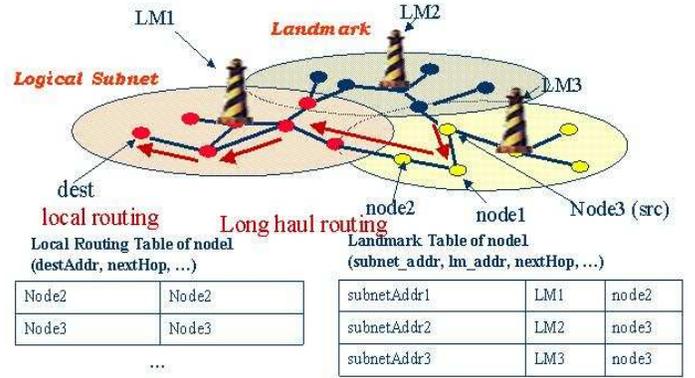
track of logical subnets. Such a logical subnet consists of nodes that have a common interest and move together as a "group". A representative of the subnet, i.e., a "landmark" node, is dynamically elected in each subnet. LANMAR protocol is supported by two complementary, cooperating routing schemes: (a) a local, "myopic" proactive routing scheme operating within a limited scope centered at each node and exchanging route information about nodes up to only a few hops; and (b) a "long haul" distance vector routing scheme that propagates the elected landmark of each subnet and the path to it into the whole network. As a result, each node maintains two routing tables: local routing table and landmark table which maintain direct routes to near by destinations and routes to all the landmarks from all the subnets respectively (see. Fig. 2). With local routing table, nodes will learn how many nodes in the same subnet are reachable from this node (i.e., can find route entry in local routing table). A node who knows more than threshold (say N) such neighbor nodes proclaims as a landmark for this subnet and broadcasts to the neighbors. When more than on mode declares itself as a landmark in the same group, the node with the largest number of group members wins the election. In case of tie, the lowest ID rules the election.

To send or relay a given packet, a node first queries a route (i.e., next hop ID) to the destination in its local routing table. With any available path, the packet will be directly forwarded to the next hop. Otherwise, this packet will be instead routed towards the publicized landmark in the same logical subnet to the destination. Note that the subnet address of the destination is carried in the packet header.

3.3 M-LANMAR Protocol
M-LANMAR (Multicast-enabled Landmark Ad hoc Routing) protocol is a proactive scheme, where group membership and multicast routes are updated proactively. With the aid of an underlying unicast protocol, the sources maintain the multicast routes to only landmarks of joined teams instead of individual paths to each member.

3.3.1 Join Multicast Group
In LANMAR, each node keeps fresh routes to all landmarks in the network by periodic landmark updates. Using the landmark updates, a team maintains its membership to multicast group(s). A landmark of a team that wishes to join the multicast group(s) *implicitly* advertises "Join Request"
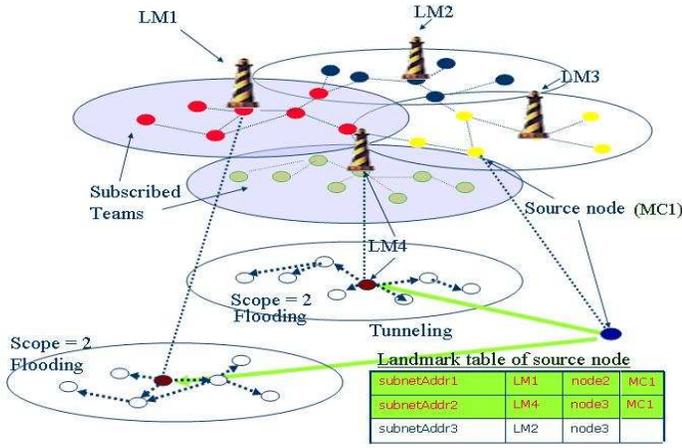
Figure 3: M-LANMAR Routing Protocol

Figure 4: Reference Point Group Mobility

to the sources by piggybacking the targeting multicast group ID(s) (address(es)) on landmark broadcast packet. Upon receiving the "implied" Join Request, each node in the network updates respective landmark entry with the subscribed multicast group IDs. Thus, the Join Request will be propagated into the sources in a few landmark table exchanges. Membership is constantly refreshed, as each landmark includes subscribed multicast addresses to all outgoing landmark update packets.

3.3.2 Leave Multicast Group
When a team who is a part of multicast group wants to leave, the landmark removes the ID of that multicast group from its subscribed multicast groups list. Thus, the landmark will stop advertising the group. The landmark's entriy at other nodes will be updated accordingly.

3.3.3 Data Propagation
The source nodes look up their landmark table to find the landmark addresses of the subscribed teams. For each landmark that subscribes to this multicast group, the source creates a "virtual link", i.e., a tunnel, to the landmark and sends encapsulated multicast data. Upon reception of the encapsulated data, each landmark initiates flooding within the subnet so that each member can receive the data (see Fig. 3) . With an assumption of restricted size of the subnet ("x" hops from the landmark to all nodes), we use local flooding with initial TTL "x+1" (in our simulation x =2). Each node in the team accepts incoming multicast data.

4. SIMULATION
In this section, we evaluate M-LANMAR. We also compare it with a robust ad hoc multicast protocol, ODMRP (On-Demand Multicast Routing Protocol) [16]. Our implementation of ODMRP follows the specification of Internet Draft (02) of ODMRP [21]. In [17], the authors showed that ODMRP generally performs well in a mobile environment compared with other MANET multicast protocols such as CAMP [15], AMRoute [7] and ARMIS [26]. With this in mind, we will limit ourselves to the comparison with ODMRP and with flooding (the latter being the most reliable scheme in a lightly loaded, mobile network). The performance of
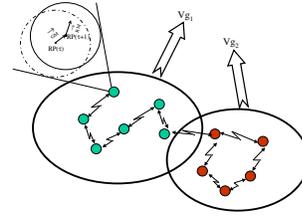
flooding, obviously, degrades as the offered load (given multicast traffic) increases [10].

We use the following metrics for our performance study:

- delivery ratio: The ratio of the number of delivered packets to each member versus the number of supposedly received packets by each member. Delivery ratio is calculated as follows:
  $(\sum_{i=1}^{N^g}((\sum_{j=1}^{N_i^m} Recv_{cnt}^j)/(Sent_{cnt}^i * N_i^m)))/N^g$, where N^g is the number of multicast group, N_i^m is the number of members of multicast group i, $Recv_{cnt}^j$ is the number of arrived multicast data at each member, and $Sent_{cnt}^i$ is the number of sent from the sources of each multicast group

- the normalized control overhead: the total number of sent control packets (e.g., Join Query/Reply in ODMRP, local/landmark routing table exchanges in M-LANMAR) is divided by the total number of delivered packets to members

4.1 Simulation Environments
We use QualNet [2] simulator, a packet level simulator developed by Scalable Network Technologies Inc. QualNet is the successor of GloMoSim [25], which provides a detailed and accurate model of the MAC and 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 Data Protocol). 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 and grouped into 36 teams. The average number of neighbors for each node is 10 and the average hop count from the landmark node to each node in the logical subnet is 2. For maintaining the routing structures, ODMRP uses 2 seconds interval for each Join Query and M-LANMAR uses 1 second interval for landmark updates and 2.3 seconds period for local routing table exchanges.

4.2 Scalability Test with Static Networks
The first experiment uses a static network to examine the scalability of the proposed idea as the number of multicast group increases. For each multicast group, three randomly selected teams join in.
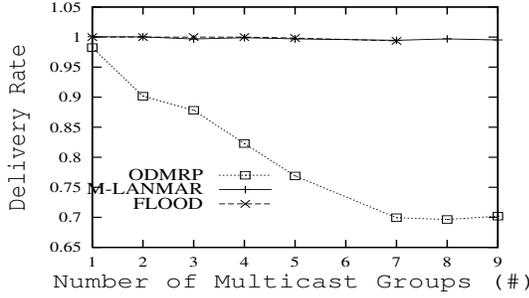
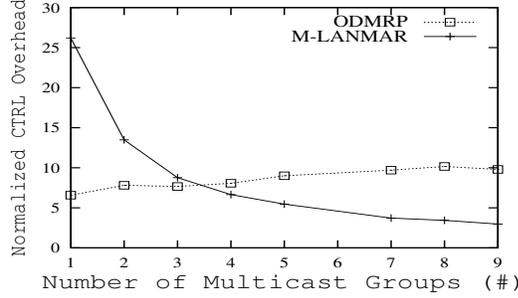**Figure 5: Delivery Ratio in the Static Network**

Figure 6: Normalized Control OH in the Static Network

Figure 7: Delivery Ratio in Mobile Network with 1pkt/sec Transmission

Figure 8: Delivery Ratio in Mobile Network with 4pkts/sec Transmission

Fig. 5 shows the delivery ratio of three protocols. This graph clearly demonstrates that the performance of ODMRP considerably drops as the number of multicast groups increases. While, M-LANMAR and FLOODING show a stable, consistently high delivery ratio. The main bottleneck of ODMRP is the excessive control overhead due to periodic maintenance messages such as Join Query and Join Reply. In MANET scenarios, due to the shared medium and limited bandwidth, the number of control overhead is extremely important and, in fact, superfluous control packets can considerably impair the delivery ratio of data [27]. Indeed, a scalable protocol should reduce protocol overhead as the offered load increases [27].

Fig. 6 shows the normalized control overhead of ODMRP and M-LANMAR. This result demonstrates that the normalized control overhead of ODMRP slightly increases as the offered load becomes heavy (i.e., the number of multicast group increases). In fact, the total control overhead of ODMRP is proportional to the number of multicast groups. On the other hand, in M-LANMAR, nodes exchange their local routing table and landmark table periodically regardless of actual offered load (i.e., M-LANMAR aggregates multicast group maintenance packets). Thus, the normalized control overhead of M-LANMAR decreases as the actual offered load increases.

4.3 Scalability Test with Mobile Networks

The following set of experiments addresses the mobile networks. A node moves following the "Reference Point Group Mobility" model [5] with speed 2m/s with 10s pause time. Fig. 4 illustrates "Reference Point Group Mobility" model. Each team moves with random vector (speed, direction) and
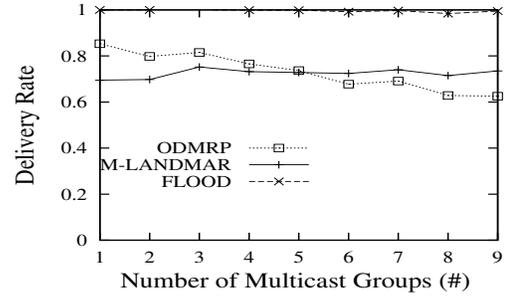
each node in the team randomly moves around the reference point. For each scenario, multiple runs with different seeds are conducted and the result is averaged over those runs. Each simulation executes for 200 seconds with randomly chosen multicast source and destination team(s). Throughout our simulation study, we use only one source node and three teams for each multicast group. The source sends out one packet every second with 512 bytes packet size as default. Here, we vary the transmission rate with 1 packet/sec and 4 packets/sec for each multicast source. We can observe three striking facts.

First, Fig. 7 illustrates that ODMRP outperforms M-LANMAR when the offered load is low. This is because with a mesh multicast structure, ODMRP provides redundant paths from the source to a destination and thus enhances the chance of packet delivery to a member even when the primary route fails. On the other hand, since M-LANMAR depends on the primary route only, the delivery ratio of M-LANMAR is considerably impaired by the route failures from the source to a landmark due to node mobility. More importantly, this failure of data transmission from the sources to a landmark leads to packet loss at all nodes in the team. This effect is similar to the case when the root of a subtree in the multicast tree fails. As a result, a single failure significantly reduces the total number of delivered packets. We also note that M-LANMAR provides, in most cases, reliable delivery to all team members once the packet has reached the landmark. In other words, if the landmark of a joined team receives the multicast data, then all members can hear the data with very high probability because M-LANMAR uses flooding within the team. In future work, we will investigate multiple paths

(redundant paths) to enhance the route availability between source and landmarks. One possible solution is to construct a mesh structure (like ODMRP) among the sources and subscribed landmarks instead of unicast tunneling.

Secondly, Fig. 8 shows that, offered load become heavier (up to four times that of the scenarios in Fig. 7), M-LANMAR does performs better than ODMRP even in presence of mobility. ODMRP suffers from heavy contention and collision due to the increase of control overhead and the number of relayed packets. Remarkably, for all scenarios, M-LANMAR shows stable delivery ratio regardless of the given offered load. All these observations put together indicate that M-LANMAR provides a scalable team multicast solution.

Lastly, the analysis of flooding shows that the delivery ratio in flooding drops with heavy offered load as shown in Fig. 8. We could not even complete the execution of the flooding runs with a large number of multicast groups (≥ 8) due to heavy memory requirements. In [18], the authors introduce the "broadcast storm" problem, where flooding results in heavy contention and collision in MANET scenarios. Indeed, with small size networks and low offered load, flooding can improve the reliability via redundant packet forwarding. However, flooding becomes inefficient due to heavy overhead in the dense and large network with high offered load. For that reason, we use restricted scope flooding in each team to exploit the advantage of flooding scheme (such as high reliability) but without paying the huge overhead.

In summary, through our extensive simulation studies, we learn that, in realistic scenarios where high offered load and large number of multicast groups are given, M-LANMAR provides an efficient and reliable team multicast solution.

4.4 Interaction with Reliable and Congestion Controlled Transport Protocol

As the large number of receivers involve in multicasting traffic, the congestion control is essential for the success of a multicast protocol. Furthermore, often applications (e.g., military applications) require reliable data transmission. The end-to-end congestion control and reliable multicasting protocols require some kinds of feedbacks from receivers (e.g., error rate, available resource, negative acknowledgement or acknowledgement). Thus, to provide the scalability, congestion control and reliable multicasting protocols should be able to work with potential feedback implosions and thus a feedback aggregation mechanism is essential. Thanks to a hierarchical approach, M-LANMAR can easily provide a platform for the feedback aggregation. As M-LANMAR reduces the number of visible receivers from outside (i.e., only landmark nodes are perceptible from the sources), it naturally makes a transport protocol scalable with group size increases.

In this section, we show the advantage of M-LANMAR platform for a reliable and congestion controlled transport protocol using RALM (Reliable Adaptive Lightweight Multicast) [1].

Interaction with RALM

In the next set of experiments, we consider the same (mo-

bile) scenario as in Fig. 8. The major protocols under consideration are still M-LANMAR and ODMRP (we dropped flooding as it exhibited poor scalability in the previous experiments). In this new set of experiments we first run ODMRP and M-LANMAR as before, with a simple UDP transport protocol on top. Then, we repeat the experiments with the reliable "multicast transport protocol Reliable Adaptive Lightweight Multicast (RALM) [1]. RALM is a reliable, rate-based, congestion controlled transport protocol that targets group operation scenarios ranging from special military operations to civilian emergency rescue applications that favor reliability over throughput. When there is no packet loss, RALM sends packets at a rate equal to the application sending rate. Once the loss is detected by a NACK (negative acknowledgement) from a receiver, RALM recovers the lost packet(s) by initiating a modified send-and-wait procedure. In RALMs send-and-wait mechanism, a sender freezes its sender buffer until the target receiver recovers all lost packets after retransmitting the recovering packets. Send-and-wait is performed with each multicast receiver that experiences the loss, one at a time in a round-robin fashion. Once all receivers have up-to-date packets, RALM reverts to the application sending rate. As one can easily observe, the number of receivers significantly affects the performance of RALM. We slightly change RALM to exploit underlying hierarchical multicast structure by M-LANMAR. With M-LANMAR, only landmark nodes (note that only landmark nodes are visible from the sources in M-LANMAR) request the packet recovery upon a data loss to the source. Other nodes recover the lost packet from their landmark nodes instead of from the source. By reducing the number of receivers in view of the sender, RALM with M-LANMAR can save the considerable amount of recovery time in the send-and-wait phase.
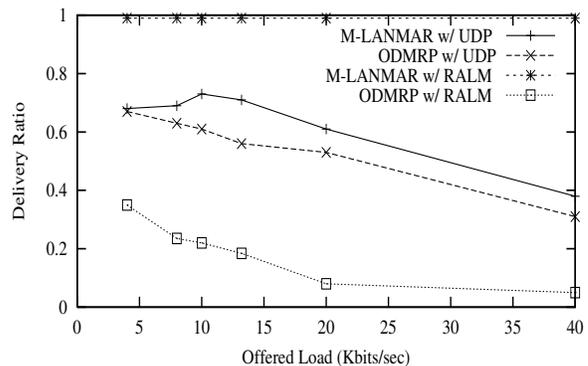

Figure 9: Delivery Ratio of ODMRP and M-LANMAR with a reliable transport protocol, RALM and with UDP

The results are shown in Fig. 9. We note that the performance of ODMRP and M-LANMAR under UDP is practically the same as reported in Fig. 8 (we show tot offered load instead of number of competing multicast groups in this case). When RALM is introduced, the M-LANMAR performance jumps to 100%. Underlying M-LANMAR significantly reduces the number of NACKs and thus RALM effectively works with M-LANMAR. ODMRP performance, on the other hand, collapses. This is due to the overhead in-
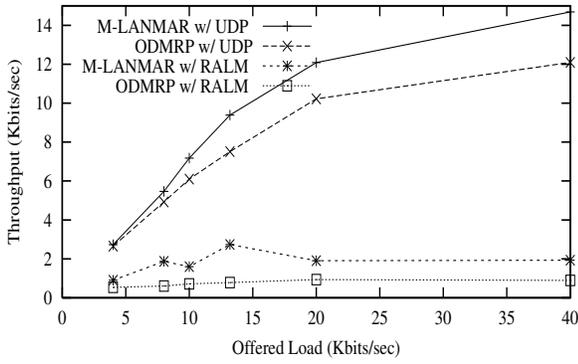
Figure 10: Throughput of each combination

troduced by the RALM protocol such as ACKs and NACKs. Clearly, RALM is not a good choice for ODMRP enhancement because of the lack of NACK suppression scheme here. In Fig. 10, we report overall throughput. We note that the "reliably delivered throughput is much lower than the aggregate throughput measured under UDP. This is expected as the stop-and-wait operation considerably slows down delivery. This is the price paid for 100% reliability (in M-LANMAR).

5. DISCUSSION
In this section, we discuss some potential extensions of M-LANMAR as our future works.

5.1 Comments on Assumptions
In the paper, for the sakes of simplicity, we assume that nodes in the same team subscribe to the same multicast groups. That assumption simplifies the intra-team forwarding scheme as scoped-flooding is effective. However, that assumption is not essential for the successful operation of M-LANMAR. Without that assumption, intra-team forwarding scheme should be re-designed since flooding can be ineffective due to redundancy and high unnecessary overhead. One easy way is to use a flat multicasting protocol such as ODMRP [16] and MAODV [20] within a team boundary to manage the multicast group membership information and propagate the data packet.

To handle member nodes of a multicast group, which are not belonging to any team, M-LANMAR can be extended to allow such nodes to receive a data packet from a near teams leader. Such nodes find a near team and ask the packet forwarding to the team leader. The team leader forwards the packet to requesters, upon receiving a new packet.

5.2 Potential Extensions
Several extensions of the M-LANMAR protocols can be explored. For instance, the unicast packet does not have to reach the Landmark of the designated team. The first node in the team that receives the packet, upon recognizing the subnet address and multicast address, can proceed to flood the packet in the subnet. This reduces latency at the expense of the reliable and congestion controlled TCP-like protocols operation.

Another extension afforded by the team model is efficient resource discovery via "content based" multicast. Consider the battlefields scenario. A tank commander wants to summon the help of UAVs (Unmanned Airborne Vehicles) in his imminent attack (he needs coverage and reconnaissance). Assuming that the landmark advertising includes also the basic team capabilities (e.g., presence of UAVs), the commander can multicast the help message to all UAV-equipped teams (as per his landmark table). The team(s) with adequate resources will respond. The commander can then choose. This is much more efficient than submitting a network wide broadcast to all nodes in the network, or maintaining a centralized (and therefore unreliable) database with all capability entries.

Moreover, M-LANMAR supports an efficient multicast QoS platform. Since M-LANMAR provides a two-tier multicasting protocol, it inherits the benefits of a hierarchical multicasting protocol. As the large receivers involve in one multicast session, QoS support for a multicast protocol should be differently designed from the unicast QoS. For a unicast QoS, the main issue is related to the resource reservation/adaptation between a source and a destination. On the other hand, a multicast QoS should consider QoS paths to all members, not only between the source and a receiver. A QoS multicast, thus, should cope with large number of receivers and be able to utilize the set of receivers. Using M-LANMAR, a hierarchical approach significantly reduces the complexity of multicast QoS protocol. Furthermore, M-LANMAR significantly contributes a heterogeneous environment where teams have different characteristics and available resources (e.g., a team consisting of tanks and a team with soldiers equipped with PDAs).

6. CONCLUSION
In this paper, we propose a new multicast paradigm, namely team multicast. We exploit the coordinated movement of teams to obtain a highly scalable multicast distribution based on the election of landmarks. As a starting point, we have implemented M-LANMAR, a landmark based scheme that uses tunneling from the source to the landmark in each team and then flooding within the team. We study the performance of M-LANMAR and compare with ODMRP and FLOOD.

From the results, we observe following three facts. First, a "flat" multicast protocol that does not exploit the affinity team model has scalability limitations. While, a team multicast protocol is well scaled as the offered load increases. Secondly, in presence of node mobility, redundant paths provided by a mesh topology can considerably enhance the delivery ratio. However, such multiple paths also exacerbate the contention and collision. Finally, team multicast not only outperforms the conventional schemes but also provides the opportunity for several enhancements such as the support of reliable delivery (e.g., via TCP), and congestion control, and resource discovery.

Since M-LANMAR uses separate tunneling from a source to each landmark, with large number of joined teams, M-LANMAR may be inefficient i.e., waste bandwidth. Thus, we will further investigate a mesh-structure like ODMRP between subscribed landmarks to improve the efficiency, ro-

bustness and scalability.

7. REFERENCES

[1]

[2] Scalable networks, http://www.scalble-solutions.com.

[3] 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.

[4] T. Ballardie, B. Cain, and Z. Xhang. Core based tree (cbt version 3) multicast routing protocol specification. *Internet draft*, March 1998.

[5] 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.

[6] 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.

[7] E. Bommaiah, M. Liu, A. McAuley, and R. Talpade. AMRoute: Ad-hoc Multicast Routing protocol. *Internet-draft, draft-talpade-manet-amroute-00.txt*, 1998.

[8] 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.

[9] 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.

[10] C. Ho, K. Obraczka, G. Tsudik, and K. Viswanath. Flooding for reliable multicast in multi-hop ad hoc networks. 1999.

[11] X. Hong and M. Gerla. Dynamic group discovery and routing in ad hoc networks. *submitted to IEEE Globecom*, 2002.

[12] 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.

[13] X. Hong, M. Gerla, Y. Yi, K. Xu, and T. Kwon. Scalable ad hoc routing in large, dense wireless networks using clustering and landmarks. *IEEE ICC*, 2002.

[14] R. Hutchins and E. Zegura. Measurement from a campus wireless network. *ICC*, 2002.

[15] J.J.Garcia-Luna-Aceves and E. L. Madruga. A multicast routing protocol for ad-hoc networks. *IEEE INFOCOM*, 1999.

[16] S.-J. Lee, M. Gerla, and C.-C. Chiang. On-demand multicast routing protocol. *Proceedings of IEEE WCNC*, 1999.

[17] 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.

[18] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Seu. The broadcast storm problem in a mobile ad hoc networks. *IEEE/ACM MOBICOM*, 1999.

[19] G. Pei, M. Gerla, X. Hong, and C.-C. Chiang. A wireless hierarchical routing protocol with group mobility. *IEEE WCNC*, 1999.

[20] E. M. Royer and C. E. Perkins. Multicast operation of the ad-hoc on-demand distance vector routing protocol. *ACM/IEEE MOBICOM*, 1999.

[21] M. G. S. Lee, W. Su. On-demand multicast routing protocol (ODMRP) for ad-hoc networks. *Internet Draft, draft-ietf-manet-odmrp-02.txt*, 2000.

[22] M. Sanchez and P. Manzoni. A java based simulator for ad-hoc networks. *http://www.disca.upv.es/misan/mobmodel.htm*, 2001.

[23] C. Shields, J. Garcia-Luna-Aceves, and P. Vallarta. The hip protocol for hierarchical multicast routing. *In Proceedings of Seventeenth Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, 1998.

[24] P. Sinha, R. Sivakumar, and V. Bharghavan. MCEDAR: Multicast core-extraction distributed ad hoc routing. *IEEE WCNC*, 1999.

[25] UCLA. Glomosim: A scalable simulation environment for wireless and wired network systems.

[26] C.-W. Wu and Y. Tay. A multicast protocol for ad-hoc wireless networks. *MILCOM*, 1999.

[27] Y. Yi, T. Kwon, and M. Gerla. The selective intermediate nodes scheme for ad hoc on demand routing protocols. *IEEE ICC*, 2002.