Energy Efficient Multicast Routing Protocol for MANET with Minimum Control Overhead (EEMPMO)

Pariza Kamboj
CSE Department, ITM University, Gurgaon, Haryana- 122017, India.

Ashok.K.Sharma
CSE Department, YMCA Institute of Engineering, Faridabad, Haryana, India.

ABSTRACT

Mobile Ad-Hoc Network (MANET) is a dynamic, multi-hop and autonomous network composed of light wireless mobile nodes. Multicast has great importance in MANET due to their inherent broadcast capability. However, due to the dynamic topology of MANETs to build optimal multicast trees and maintaining group membership a lot many control messages required. These overhead consume the mobile node resources like power and network resources like wireless links bandwidth that creates hurdle in implementing energy assurance and reduced overhead multicast protocol for Mobile Ad hoc Networks (MANET). This paper presents an energy efficient multicast routing protocol for MANET with minimum control overhead. The protocol creates shared multicast tree using the physical location of the nodes for the multicast sessions. Protocol employs a distributed location service to obtain the physical location information of the nodes, which effectively reduces the overheads for route searching and shared multicast tree maintenance. The algorithm uses the concept of small overlapped zones around each node for proactive topology maintenance with in the zone. To search for an existing multicast tree outside the zone, constrained directional forwarding is used which guarantees a good reduction in overhead in comparison to network wide flooding for search process. In this paper local connectivity technique and preventive route reconfiguration on the basis of the current status of the nodes are being proposed that attempts to improve the performance and reliability in terms of reduced overhead, power and bandwidth requirement. These techniques also ensure good reduction in latency in case of link breakages and prevention of the network from splitting.

General Terms

Topological routing protocol, tree based multicast, mesh based multicast, shared-tree based multicast protocol, source-tree based multicast protocol.

Keywords

Mobile ad-hoc networks, multicasting, routing zone, shared trees, physical location, geographic location service GPS, preventive route reconfiguration, grid location service GLS.

1. INTRODUCTION

An ad hoc network consists of a collection of mobile hosts forming a dynamic multi-hop autonomous network [1] without the intervention of any centralized access point or fixed infrastructure. Multicast has great impact in mobile networks because of their inherent broadcast capability. Using multicast instead of sending through multiple unicasts not only minimizes link consumption, but also reduces sender and router processing, communication costs and delivery delay [2].

Group communication is important in Mobile Ad Hoc Networks (MANET). Many ad hoc network applications which require close association of the member nodes depends on group communication. Action directions given to the soldiers in a battlefield and communications required during a rescue operation are some examples of these applications. In addition, many routing protocols for wireless MANETs need a broadcast/multicast as a communication primitive to update their states and maintain the routes between nodes [3].

Multicast protocols can be categorized in tree based and mesh based protocols. Multicast network structures are frail therefore need to be readjusted and repaired continuously as the connectivity changes. Multicast protocols have to produce multi-hop routes under bandwidth scarcity, limited battery power and dynamic topology due to nodes' unpredictable mobility. Even in wired networks, building optimal multicast trees and maintaining group membership information is challenging which becomes predominantly challenging in mobile ad hoc networks.

The proposed protocol, an energy efficient multicast routing protocol for MANET with minimum control overhead, called EEMPMO uses the concept of proactive zone and constructs a shared bi-directional multicast tree with back up root for its routing operations. Zone building, multicast tree construction and multicast packet forwarding depends on the location information obtained using a distributed location service GLS, which effectively reduces the overheads for route searching and shared multicast tree maintenance. To search for an existing multicast tree outside the zone, constrained directional forwarding is used which guarantees a good reduction in overhead in comparison to network wide flooding for search process. Performance and reliability in terms of reduced overhead, less consumption of power and bandwidth is improved using the local connectivity technique and preventive route reconfiguration on the basis of the current status of the nodes. These techniques also ensure good reduction in latency in case of link breakages and prevention of the network from splitting.

The rest of the paper is organized as follows: Section 2 describes tree based multicast protocols classification for MANET and also emphasizes the problems lie in the existing multicast routing protocols. The proposed energy efficient multicast routing protocol for MANET with minimum control overhead is discussed in Section 3. Section 4 analyses the performance of EEMPMO in comparison with other shared-tree based multicast protocol MAODV. Finally, section 5 summarizes the study of the work in conclusions.

2. MULTICAST PROTOCOLS FOR MANETS

Most of the multicast protocols proposed for mobile ad hoc networks can be broadly categorized into two types, namely tree-based multicast and mesh-based multicast. Multicast mesh does not perform well in terms of energy efficiency due to excessive overhead as it depends on broadcast flooding within the mesh. On the other hand tree structure is known for its efficiency in utilizing the network resource optimally which is the motivation behind the selection of tree based multicast. A tree based multicast routing protocol can be either a source-tree or a shared-tree based protocol. Multiple source-tree based routing trees routed at different sources of the multicast session are used to deliver data packets in a source-tree based multicast routing protocol while a shared multicast tree for the whole multicast group is used to deliver data packets in a shared-tree based multicast routing protocol. Source-tree based multicast cause excessive overhead to reconstruct a large number of source trees in case of highly mobile nodes [4], while shared tree multicast has lower control overhead because it needs to maintain only a single shared tree for all multicast sources and therefore is more scalable [5].

2.1 Comparison of Multicast Protocols

Ad hoc Multicast Routing (AMRoute) [6] and Lightweight Adaptive Multicast (LAM) [7] are tree based protocols, in which a shared tree is constructed for the delivery of multicast packets to the entire multicast group. In AMRoute protocol a bi-directional shared multicast tree is created involving only the group members. The tree links are created as unicast tunnels between the tree members. The problem with AMRoute is that it depends heavily on an underlying unicast protocol for creating these unicast tunnels. The LAM protocol depends on Temporally-Ordered Routing Algorithm (TORA) [8] for route finding ability and cannot operate independently. An advantage of LAM is that, it reduces the amount of control overhead generated for route finding, due to its tight coupling with TORA. CAMP [9] and On-Demand Multicast Routing Protocol (ODMRP) [10] are well-known examples of mesh-based multicast routing protocols. They enhance the robustness by providing redundant paths between the source and destination pairs. The mesh is created at the cost of higher forwarding overhead. CAMP illustrates a proactive mesh based protocol. On the other hand, in ODMRP, the mesh is created using the forwarding group concept and a reactive approach is followed to keep the forwarding group current [4].

The main disadvantage with mesh based protocols is the excessive overhead incurred in keeping the forwarding group current and in the global flooding of the JOINREQUEST packets. Even the shared tree approach has some other drawbacks:

(i) Due to shared tree structure these protocols have the disadvantage of their dependency on a core node to maintain group information and to create multicast tree, thus have a central point of failure.

(ii) Due to node mobility the tree structure is fragile and thus, need updation. To compensate this problem and to optimize the multicast tree, multicast protocols for MANETs usually employ control packets to periodically refresh the network structure [11], which causes increments in the overhead and power consumption.

(iii) Every multicast routing protocol is having some or the other problem, hence suitable to specific kind of environment.

To alleviate the problem of dependency on a core node, a back up root node along with the primary root node is used. To reduce overhead and power requirement, the constrained directional forwarding in the direction of the target using its location information employed in the protocol. To make an environment independent protocol, a hybrid approach is the need of the protocol. Moreover, the location advantage of the nodes can further improve the performance of the protocol manifolds. Based on this view we have designed a new multicast routing protocol named Energy Efficient Multicast Routing Protocol for MANET with Minimum Control Overhead (EEMPMO).

3. PROPOSED PROTOCOL EEMPMO

This section introduces a new multicast protocol, Energy Efficient Multicast Routing Protocol for MANET with Minimum Control Overhead, which follows a hybrid approach using the grid location service to gather the physical location of the nodes. Use of backup root node provides support in case of primary root node failure. The protocol reduces the total energy consumption as well as improves the performance than a conventional shared tree based protocol by reducing the overhead.

3.1 Shared Multicast Tree with Backup Root

In case of shared multicast tree the protocol dependency on a root node to maintain the group information burdens the root node. Due to this shared tree multicast is particularly not suitable from energy balancing point of view because the root of the tree takes on more responsibility for routing, consumes more battery energy, and stops working earlier than other nodes. This leads to reduced network lifetime [12] and the whole multicast tree is disconnected into a number of partitions which consumes a lot of wireless bandwidth for reconstructing the multicast tree from all these partitions. To alleviate this problem, EEMPMO creates the shared multicast tree with backup root node as an alternative to the primary root node. Creation of a backup root node enhances the performance of the multicast tree and also lessens the load on the primary root node. In case of primary root node failure the backup root node takes over, therefore, reduces the dependency on a single root node. This facilitates a great reduction in tree maintenance and tree re-construction overhead. Selection of backup root node is done from the neighbor nodes of the primary root node on the basis of stability, battery status and proximity. A non-tree member node with slow movement and more power status is chosen to be the backup root node. If the root node does not found any neighbor node with the required criterion then the selection process is delayed by some random time and after that the backup root node search process starts again. The selection process may lead to slight delay but improves overall efficiency of the protocol by selecting a suitable node as backup node. Selecting a suitable node as backup root node not only serves the purpose of standby root node but also defer the early possibility of searching the backup root node again in case of power failure or movement of the existing backup root node.

Figure 1 shows an example of a multicast tree. The tree consists of a primary root node (R), backup root node (B), three intermediate nodes (I), six member nodes of a multicast group, and nine tree links. A multicast packet is delivered from the root node R to all the six group members. Using the zone routing every tree member unicasts the multicast packet only to the neighbor tree members, thus saves a lot many transmissions otherwise required in case of broadcasts.

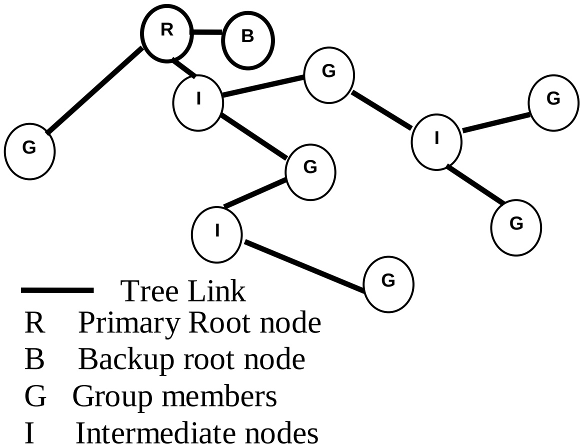

Fig :1 Shared Multicast Tree with Backup Root

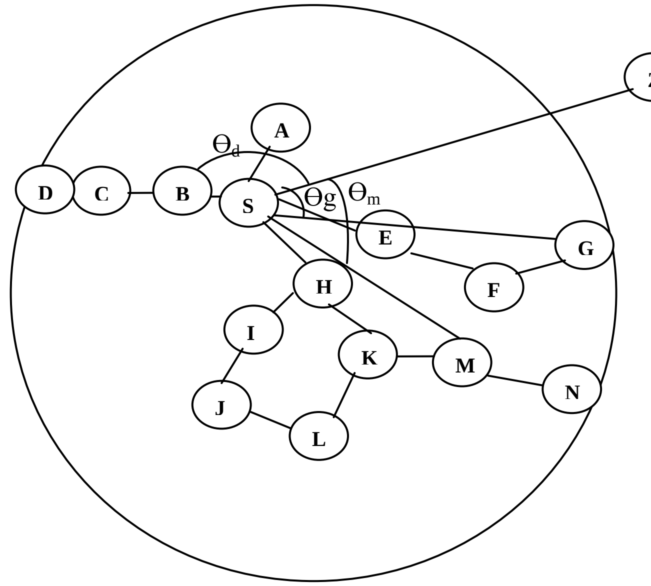

Fig.2. 3 hop Routing zone at node S

3.2 Zone Routing

A routing zone is defined for each node separately, and the zones of neighboring nodes overlap. A k-hop routing zone of node S can be defined as a connected topological subgraph, on which node S is aware of the route to any other node [13]. The nodes of a zone are divided into border nodes and interior nodes. Border nodes are nodes which are exactly k hops away from the node in question. The nodes which are less than k hops away are interior nodes. In fig. 2, the nodes G, D and M are border nodes and rest all are interior nodes and the node N, 4 hops away from S, is outside the routing zone. However node L is within the zone, since the shortest path up to L with length 3 is less than the maximum routing zone hops.

To manage the overhead, the proactive scope is reduced to a small zone around each node in the EEMPMO protocol. As the zone radius is significantly smaller than the network radius, the cost of learning the zones' topologies is a very small fraction of the cost required by a global proactive mechanism. Zone routing is also much cheaper (in terms of control traffic and congestion) and faster than a global reactive route discovery mechanism, as the number of nodes queried in the process is very small [4]. A bigger proactive zone can be selected for comparatively stable topology where the updates of topology are done on topology change only. In a limited zone, each node maintains a proactive unicast route to every other node. In the proposed protocol the routing is initially established with proactively prospected routes within the zone and then outside the zone, using diffused routing towards the tree members. Therefore, route requests can be more efficiently performed without exploiting the flooding in the network.

3.3 Physical Location of Mobile Nodes

The routing performance can be significantly improved by utilizing location information of nodes in communication e.g., if a sender node knows the location of the tree member, it can find out the route to the tree member using constrained routing by forwarding the packet in the relative direction in hopes of getting it there quickly therefore communication delay can be minimized with location information [14]. A node can use Global Positioning System (GPS) to obtain its geographic location information. The locations of other nodes can be obtained by employing some distributed location service. However, in practice, it is difficult to find/maintain node locations with accuracy in an ad hoc environment where nodes move around. Some well-known location-based routing algorithms are location-aided routing (LAR) protocol [15], distance routing effect algorithm for mobility (DREAM) [16] and grid location service (GLS) [17]. DREAM is a global proactive location service as it flood position updates in the whole network proactively which is used by all nodes in the network to build the complete position data base. This scheme causes a lot many overhead and also big requirement of the memory on all nodes. On the contrary, LAR is global reactive location service that causes a pretty long delay for the location updates at far away nodes and also the overhead due to global flooding. Due to this both the schemes are not suitable in terms of the network congestion and overhead, therefore EEMPMO uses a grid location service to provide location information to all mobile ad hoc nodes and the geographical information thus obtained is used to limit the flooding of packets to a small region.

3.4 EEMPMO's Modified Data Structures

GLS represents a fully distributed and scalable location service in the following manners: No node is a bottleneck as responsibility of maintaining the location service is spread evenly over all the nodes. Failure of a node does not affect the reachability to many other nodes. Local communication satisfies the queries for the locations of the nearby nodes which also allow operation in the face of network partitions. The communication cost and per-node storage of the location service grow as a small function of total number of nodes [17]. GLS employs a number of nodes as "location servers" distributed throughout the network, which provides location information to other nodes. Although each node has the ability to act as a location node, EEMPMO prefers a node rich in resources like memory and comparatively stable to be location node. In order to facilitate the location service, each node has some data structures in addition to those needed for the routing algorithm. The data structures used in EEMPMO are amended ones and in addition to the existing ones to improve the performance of the routing.

Each node maintains a localized "Location Table (LT)" to keep the record of the neighbors within k-hop zone as shown in table 1. Each routing entry contains the IP of neighbor node, location, speed, next immediate hop towards that node, total hop counts to reach to this node and a timestamp indicating when the entry was added or updated. In case of extra space available in a node, it may store the entries of other nodes in addition to its zone neighbors about which it learnt by passively listening on the network and with which it communicates. Entries expire from the table after a certain time period, in order to clear a node's table of possibly outdated information. The number of entries stored in the location table is related to the node's "goodness" score, described below.

The second data structure that each node maintains is the "scorecard" of other nodes as shown in table 2. This is a table where each entry contains the IP of a node and a score indicating how "good" the node is at providing location information to the nodes outside its zone. Entries are made in the descending order of the score values and only of those nodes having a score value more than a threshold sth. These nodes represent the location servers. A small value score indicates a bad location node in providing location information to other nodes, while a high value for the score indicates that the node stores more nodes' locations. It may be initialized proportional to the available size of the node's location table. When the node answers a source node's request, score is increased and when the node moves more than dth distance from its original place, its entry is removed from the scorecard table. Score is decreased over time through a score decay mechanism. When the score decreases than sth, the entry will be removed from the table. The reason for this score decrement is to prevent nodes from expending energy that rarely provide locations, even if they have large capacities.

When a source node needs the location of a target outside, it consults its scorecard and sends a MGREQ request to the highest-scoring location node. If a response is not heard after a certain amount of time, the node's score is decreased and the source node asks the next highest-scoring node. When a response is received, the source node increases the node's score. The amount by which a score is increased should reflect how long a node takes to answer a request, and how up-to-date the location information received from the node is.

Besides LT and scorecard, for the purpose of routing information each node maintains Multicast Tree Table (MTT) as shown in table 3 and Request Table (RT) as shown in table 4.

Each entry of multicast tree table contains the multicast group IP address, multicast group leader IP address, hop count to multicast group leader, next hops and timestamp. This table has entries for all those multicast groups of which group the node is a member. The Next Hops field is a linked list of structures, each of which contains the following fields:

- Next Hop IP Address
- Link Direction
- Activated Flag

The direction of the link is relative to the location of the group leader. UPSTREAM is a next hop towards the group leader, and DOWNSTREAM is a next hop away from the group leader [13]. An entry is added to the table when the node becomes a multicast group member.

A request table is maintained by all those nodes that support multicast. An entry in this table contains multicast group IP address, tree member (requesting node) IP address, tree member node's location and a timestamp. On reception of MGREQ from a request node for a multicast group an entry is made in this table.

In order to exchange location information on the network, four special packet types are exchanged. A HELLO packet as shown in fig. 3 is broadcasted by a node within its zone only rather than flooding in the whole network, when it wants to inform other node(s) of its location. It contains the IP, location (latitude and longitude) of the source node, speed of the source node and a timestamp.

In response to the HELLO packet the receiving node unicasts back an acknowledgement packet ACK as shown in fig. 4. This packet contains the IP and location of the source node, the IP and location of the node acknowledging receipt of a HELLO and a timestamp.

For the purpose of finding the distance d between two nodes equation (1) is used and slope θ made by a line joining the source node and peripheral node with the line from source to member node is calculated using (2).

$$d = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \quad (1)$$

$$\theta = \tan^{-1} \frac{(y2 - y1)}{(x2 - x1)} \quad (2)$$

where (x1,y1) and (x2,y2) are the locations of two mobile nodes.

Table 1: Location Table maintained by nodes

IP	Location		Speed (m/s)	Next Hop	Total hops	Timestamp	Partiton ID
	Latitude	longitude					
222.24.15.06	420 10' E	560 40' S	5	222.24.15.15	3	15:09 PM	2,2
222.24.15.11	550 10' W	340 33' S	7	222.24.15.31	2	15:15 PM	3,2
222.24.15.20	230 26' E	150 14' N	8	222.24.15.19	3	15:24 PM	3,4
222.24.15.29	450 30' N	430 20' E	9	222.24.15.43	1	15:42 PM	1,3

Table 2: Scorecard maintained by each node

IP	Score
222.24.15.06	35
222.24.15.11	30
222.24.15.20	27
222.24.15.29	12

Table 3: Multicast Tree Table

IP Multicast group MG_IP	IP Multicast Group Leader MGL_IP	Hop Count HOP_CNT	Next hop NXT_HOP	Time-stamp TS
224.30.15.10	222.24.15.50	8	222.24.15.05	15:39 PM
224.30.10.10	222.24.15.65	9	222.24.15.11	15:45 PM
224.30.10.15	222.24.15.36	7	222.24.15.10	15:04 PM
224.30.10.50	222.24.15.45	10	222.24.15.20	15:12 PM

Table 4: Request Table

IP Multicast group MG_IP	IP Multicast group Tree Member TM_IP	Location of Multicast group Tree Member		Time-stamp TS
		Latitude TM_LAT	longitude TM_LNGT	
224.30.15.10	222.24.15.06	420 10' E	560 40' S	13:41 PM
224.30.10.10	222.24.15.11	550 10' W	340 33' S	14:38 PM
224.30.10.15	222.24.15.20	230 26' E	150 14' N	13:24 PM
224.30.10.50	222.24.15.29	450 30' N	430 20' E	14:12 PM

Source Node			Velocity SRC_V	Timestamp TS
IP SRC_IP	Latitude SRC_LAT	Longitude SRC_LNGT		

Figure 3. Format of HELLO packet

Ack. Node			Source Node			Time-stamp TS
IP ACK_IP	Latitude ACK_LAT	Longitude ACK_LNGT	IP SRC_IP	Latitude SRC_LAT	Longitude SRC_LNGT	

Figure 4. Format of ACK packet

Request Node			Multicast Group IP	Join Flag	Time-Stamp
IP RQ_ IP	Latitude RQ_ LAT	Longitude RQ_L NGT	MG_IP	JF	TS

Figure 5. Format of MGREQ packet

Multicast Group Tree Member			Request node			Time stamp
IP	Latitude	Longitude	RQ_	Latitude	Longitude	
TM_IP	TM_LAT	TM_LNGT	IP	RQ_LAT	RQ_LNGT	TS

Figure 6. Format of MGRPL packet

A multicast group request packet MGREQ, shown in fig. 5, is broadcasted by a node within its zone in search of an existing multicast group. This packet contains the IP and location of the request node, IP of the multicast group, join-flag and a timestamp. A location reply packet MGRPL as shown in fig. 6 is sent in response to a MGREQ packet by a tree member node. The MGRPL packet contains the IP and location of the multicast group tree member, the IP and location of the request node, and a timestamp.

3.5 Neighborhood Connectivity Updation

Nodes learn of their neighbors through transmission of HELLO, ACK, MGREQ and MGRPL packets. In EEMPMO, a node broadcasts HELLO packet periodically to inform other node(s) of its location with TTL value equal to k hops whenever it enters into a network or whenever it moves significantly (i.e. equal to dth) from the previous location calculated as (1). The receiving neighbor nodes unicast the ACK packet back to the sending node to get update their locations. A node also learns of its neighbors by promiscuous snooping on the channel for detecting activities of neighbors.

3.6 Shared Tree Creation

EEMPMO maintains a bi-directional shared multicast tree for each multicast group, consisting of the members of the multicast group and several routers. Each multicast group has a unique multicast group address (IP) [18] and a group leader. The group member that first constructs the tree is designated as the group leader or the primary root of the tree [19]. EEMPMO algorithm searches the multicast group tree member in the zone of the intermediate node. It searches the possibility of tree member by searching the multicast group IP in the multicast tree table and request table of each neighbor node in the zone. In case of no match found with in the zone it repeats the search outside the zone.

3.6.1 Searching the existing multicast group in zone -

Proactive topological routing operates within the k-hop routing zone. A request node, that wants to join the multicast group, will first look for the existing tree of the multicast group. The node broadcasts a MGREQ packet with multicast group IP and join flag set within its k-hop routing zone (TTL=k). All nodes of the zone search the multicast group IP in their multicast tree table. A node having a matched entry replies back MGRPL unicastly to the request node by putting its own IP, latitude and longitude in the multicast tree member IP (TM_IP), latitude (TM_LAT) and longitude (TM_LNGT) fields of the MGRPL through the reverse route maintained during the traversing of MGREQ packet. In case of no entry matches in the multicast tree table of all the neighbor nodes, the request node searches the tree existence outside the zone.

3.6.2 Searching the existing multicast group outside zone

To find the possibility of the group outside the zone, the multicast IP of the MGREQ packet is searched in the request table of the zonal nodes. If any entry of the request table matches, then the node unicasts MGRPL to the request node by putting the IP, latitude and longitude of the matched entry node in the multicast tree member IP, latitude and longitude fields of the MGRPL. The matched entry node, in the request table, indicates a node that had requested for the multicast group in the past and hence actually the tree member node outside the zone.

In case of no entry matches in the multicast tree table and the request table of the nodes in the zone, the node finally checks its scorecard and sends the MGREQ packet to the highest scoring node of its scorecard. It waits for a certain amount of time and in case of no response it sends the packet to the next highest scoring node. Continuing in this way it enquires from all the nodes of the scorecard. Still in case of no success, the request node sends a small signal to its border nodes like to forward the cached copy of the MGREQ to all the border nodes of their respective zones in search of the multicast group in the whole network like D, G, J, L and M as shown in fig. 2. In case of no border nodes, signal is sent to the nodes with k-1 hop like C, F, I and K in fig. 7 and to all possible nodes in case of sparse network. The border nodes then broadcast the MGREQ packet to all the nodes in their zone. These nodes further search the multicast group IP in their multicast tree table and request table until the network diameter is not reached. A node having a matched entry in either table replies MGRPL back unicastly to the request node.

3.6.3 Confirm the join process

After receiving the MGRPL the request node broadcasts a stop search signal to all nodes in its zone and sends the GRAFT message to confirm the join process following the forward route to the node from which it received the MGRPL message. The GRAFT message will activate the tree link between the request node and the node which sent the MGRPL message and this way the request node becomes the tree member. Request node also updates its request table and multicast tree table.

3.6.4 Creating a new tree for a new multicast group -

Once the whole network is traced (TTL equals Network_Diameter) in search of multicast group and still no

MGRPL is received by the request node, it assumes that the requested multicast group does not exist. It then declares itself the leader of the multicast group and becomes the primary root of the tree and broadcasts this information to all nodes within its zone.

3.7 Shared Tree Preventive Updation

The robustness of the multicast tree is adversely affected with the time if individual links are repaired only when broken. Over a period of time due to high mobility of the nodes the overall structure of the tree would be far from optimal, hence making the tree susceptible to even more link breakages. In EEMPMO, the tree is updated regularly and also the preventive maintenance is done which kept the tree robust.

3.7.1. Tree Updation

In order to maintain the tree structure even when nodes move, group members periodically send tree_update requests to the backup root node to lessen the load on the primary root node. The multicast tree can be updated using the path information included in the tree_update request messages. If any change is found in the path the back up root node sends an update message to the primary root node to notify about the change so that the changes in the topology also reflect in the tree structure. Tree_update need to be initiated by leaf nodes only as each uplink next hop puts its own uplink on the tree update message, therefore contains all uplinks as it travels towards backup root node. The period must be carefully chosen to balance the overhead associated with tree update and the delay caused by the tree not being timely updated when nodes move [6, 18, 20].

3.7.2 Preventive Maintenance - Preventive approach is being used for tree reconstruction prior to link breakages in case the tree member wants to leave the tree or its power resource is going to deplete.

A non-leaf node wishing to move out of the multicast tree, will broadcast an alarm message with TTL value 1 to its neighbors before sending the Leave message. It then compares the distance of nodes in its LT and passes all of its routing information to a nearest node which is not a tree member. New links are grafted on the tree from the upstream node and downstream nodes of the leaving node to the newly found neighbor node. The downstream node sends tree_update to the backup root node. All the future transmissions follow the path with newly discovered link. In case of leaf node or a normal network node, the node simply sends the leave message to its one hop neighbor nodes. All the neighbor nodes receiving the alarm packet from any node also remove the related entry from their LT and also from request table, if the entry with IP of leaving node exists there. In case of primary root mobility, the primary root sends the alarm message to back up root notifying it to take the control of the tree and passes its all routing information to the back up root. Upon receiving the alarm message, the back up root updates its downstream next hops to the downstream next hops of the primary root node. It also selects a new back up root for its replacement after it resumes as primary root node.

In case of the depletion of the battery power of a tree member node, link is repaired prior to its breakage. The battery power of the nodes in the multicast tree is examined periodically (frequency of examination is doubled in case of primary root node) and if the power source of a node goes below a threshold value, a new link is discovered prior to its failure, and the links to this node are deleted from the multicast tree. New link is searched in the same way as in case of leaving the tree process.

The latency in finding new route in case of nodes failure is reduced by reconfiguring the routes using preventive approach before the failure of the node.

3.7.3 Tree Repair

When a link breakage is detected, the downstream node of the break (node farther away from the group leader) initiates to repair the link by broadcasting a MGREQ-J within the zone. Only a tree node with lesser hop count to the leader (that is nearer to the group leader) may respond to this MGREQ. If the node receives a reply it then grafts a new branch using GRAFT message up to the node which sent the MGRPL.

3.8 Constrained Directional Forwarding for Data Transmission

After receiving the MGRPL the request node then sent the data packet to the tree member along the forward route created during MGRPL transmission, if the tree member node is found through multicast tree table of any node inside or outside the zone. If the tree member node is found through request table of any node within or outside the zone then a route up to the tree member is found out by constrained directional forwarding using its location information. Constrained directional forwarding algorithm forwards the same MGRPL having the IP and location of the tree member, to the border nodes or the farthest nodes in the zone in the direction of the specified tree member. The request node selects only those nodes lying on the perimeter of its k-hop zone whose slope with the direction of tree member is less than the threshold value of the slope, hence geographically closer to the tree member. Slope can be find out by (2) using the latitude and longitude of the nodes. As shown in fig. 2, node S selects the border nodes G and M only as their slope magnitude with the direction of tree member, θg and θm are less than the threshold slope magnitude θt while the slope magnitude of D i.e. θd is more than θt. After selecting the nodes, the MGRPL packet is forwarded to the next hops towards the selected border nodes. In case of sparse network, if there is no border node in the zone then the MGRPL packet is forwarded to only those farthest neighbor nodes in the zone which are having slope less than the threshold slope with the direction of the tree member.

As shown in fig. 7, node S forwards the packet towards the neighbor nodes F and K through next hops E and H only as their slope magnitude θf and θk are less than the threshold slope magnitude θt while the slope magnitude of C i.e. θc is more than θt. If no such nodes are found then the MGRPL packet is forwarded to all neighbor nodes within the zone which further forwards the packet to their border nodes in the direction of the tree member as described above.

As shown in fig. 8, node S forwards the packet towards all the neighbor nodes C, F and H through next hops B, E and H as no node is having the slope less than the threshold slope magnitude θt. Thereafter these border or farthest nodes will forward the route search packet to the border nodes of their respective k-hop zones in the direction of tree member only.

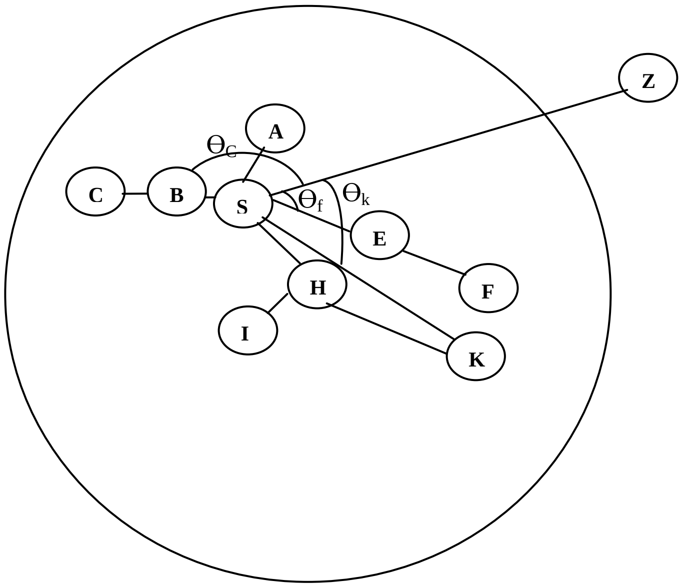

Fig. 7. 3 hop Routing zone at node S with no border nodes

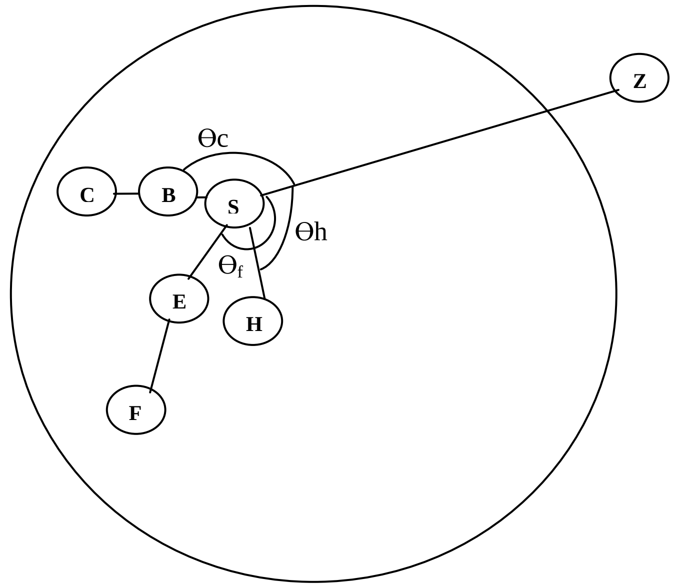

Fig.8. 3 hop Routing zone at node S with no slope condition satisfying node

This process goes on until the packet reaches to the tree member specified in the MGRPL packet. After accepting the first copy of MGRPL packet rest copies are discarded by the tree member. This tree member now replies back the same MGRPL to the request node as confirmation. This way a route is confirmed from the request node to the tree member node. Finally the request node transmits the data packet to the tree member along the forward route created this way.

Since the traffic would be forwarded only through limited nodes to tree members for route discovery using constrained directional forwarding it effectively reduces the traffic and saves the bandwidth a lot.

4. PERFORMANCE COMPARISON

4.1 Simulation Testbed

For the simulation of the protocol NS-2.26 simulator has been used. The nodes use the IEEE 802.11 radio and MAC model provided by the CMU extensions. The nodes are placed at uniformly random locations in a square universe. We generate 50 mobile hosts moving randomly within a flat square (1000m X 1000m) area. The model is configured with 100 pixels radio transmission power and 2 Mb/s basic data rate as a sample case. Two Ray Ground mobility model with node speed of 10m/s was used for the simulation. Each simulation was run for 900 simulated seconds. Data traffic was generated using constant bit rate (CBR) UDP traffic sources with 5, 10, 15, 20 and 25 mobile nodes acting as receivers in the multicast group. The node chooses a random destination and moves toward it with a constant speed chosen uniformly between zero and a maximum speed (10 m/s).

4.2 Performance Metrics

The metrics used for performance evaluation were: (i) Consumption of power of the nodes in the network. (ii) Average end-to-end delay of data packets - this includes all possible delays caused by buffering during route discovery, queuing delay at the interface, retransmission delays at the MAC, propagation and transfer times. (iii) Packet delivery ratio — the ratio obtained by dividing the number of data packets correctly received by the destination by the number of data packets originated by the source. (iv) Overhead – this includes control overhead required for tree re-construction, maintenance and route search process. Figures compare the performance of EEMPMO with that of MAODV as a function of no. of receivers. Comparison of energy consumption is shown in fig. 9, end-to-end delay in fig. 10, delivery ratio in fig. 11 and overhead generated of EEMPMO and MAODV protocols is shown in fig. 12. In all respects the EEMPMO outperforms MAODV due to the constrained directional forwarding in the direction of the target only instead of exploiting the broadcast in the whole network. Location information obtained through grid location service is very useful in this regard.

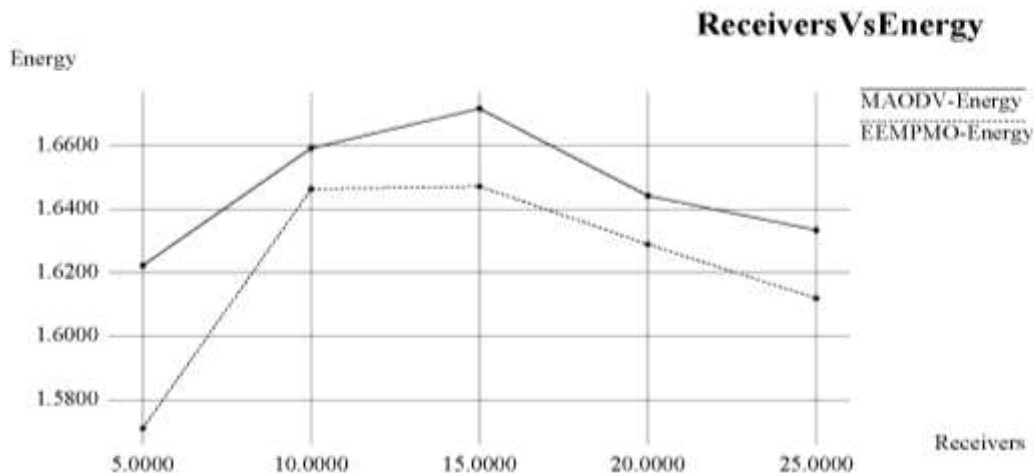

Fig. 9. Energy Consumption as a Function of Receivers

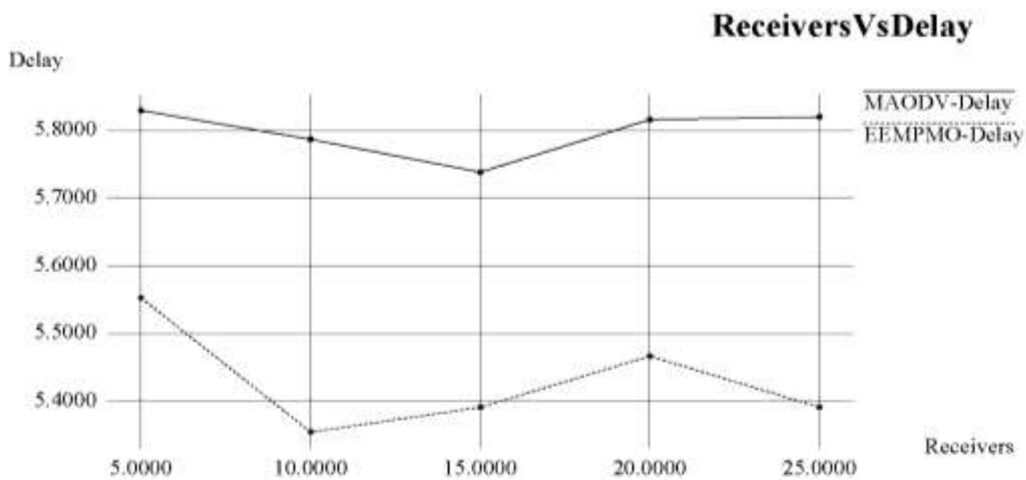

Fig. 10. End-to-End Delay as a Function of Receivers

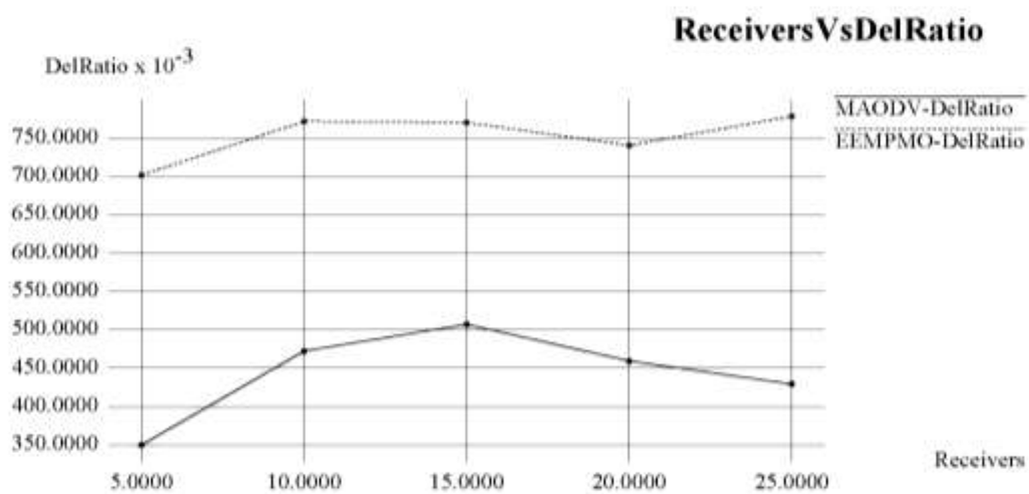

Fig. 11. Delivery Ratio as a Function of Receivers

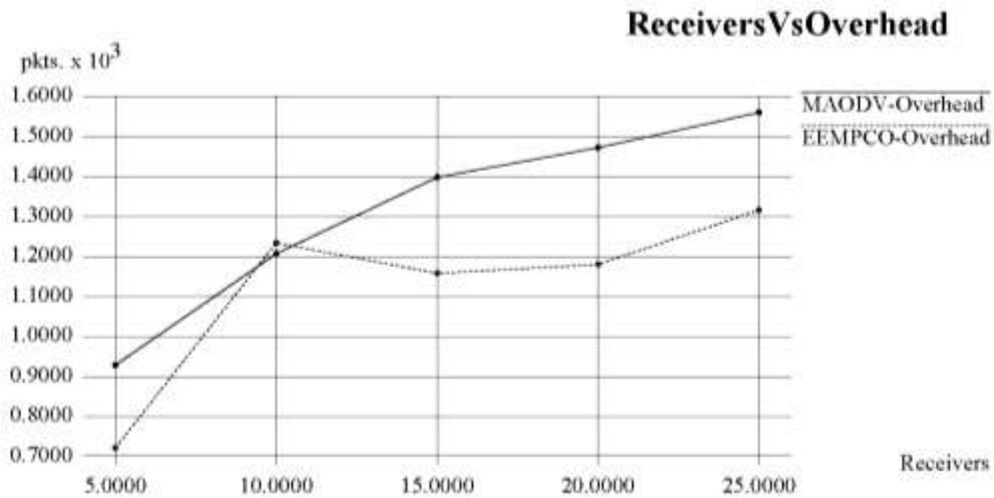

Fig. 12. Overhead as a Function of Receivers

5. CONCLUSION

The Energy Efficient Routing Multicast Protocol for MANET with Minimum Control Overhead is compared with other shared tree multicast protocol i.e. MAODV. Comparison was made on various parameters like Energy Consumption, Packet Delivery Ratio, Delay, and Throughput.

EEMPMO eliminates the drawbacks even of the shared tree protocols. It reduces the delay problem due to directional diffused forwarding routing and also the network partition problem when a link error occurs due to the failure of primary root. Due to the physical location of the nodes obtained through GLS the route finding process becomes faster, therefore the packets are delivered on a fast pace.

Backup root also facilitates reduction in overhead in case of EEMPMO otherwise required for tree reconstruction and tree maintenance. This result in improved packet delivery ratio and energy balance compared to the conventional shared tree multicast (STM) due to preventive maintenance and also because of support from the backup root in case of primary root failure.

Scalability is achieved due to the shared tree multicast routing protocol as single tree maintenance for all group members is easier than the maintenance of number of trees in case of source based multicast routing protocol.

6. REFERENCES

[1] A.K. Sharma and Amit Goel, "Moment to Moment Node Transition Awareness Protocol (MOMENTAP)", International Journal of Computer Applications (IJCA) Special Issue, IASTED, Vol. 27/1, Jan 2005, pp. 1-9.

[2] Stephen Mueller, Rose P. Tsang and Dipak Ghosal, "Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges".

[3] Song Guo, Member, IEEE, and Oliver Yang, Senior Member, IEEE, "Maximizing Multicast Communication Lifetime in Wireless Mobile Ad Hoc Networks", IEEE Transactions on Vehicular Technology, vol. 57, no. 4, July 2008.

[4] Aniruddha Rangnekar, Ying Zhang,Ali A. Selcuk, Ali Bicak, Vijay Devarapalli, Deepinder Sidhu, "A Zone-Based Shared-Tree Multicast Protocol for Mobile Ad Hoc Networks", In Vehicular Technology Conference, 2003, 2003.

[5] M. Gerla, C. Chiang, and L. Zhang, "Tree Multicast Strategies in Mobile, Multihop Wireless Networks," Baltzer/ACM Journal of Mobile Networks and Applications (MONET), Vol. 3, No. 3, pp. 193-207, 1999.

[6] M. Liu, R. R. Talpade, A. McAuley, and E. Bommaiah, "AMRoute: Adhoc Multicast Routing Protocol," Technical Report, vol. TR 99-8, The Institute for Systems Research, Univesity of Maryland, 1999.

[7] L. Ji and M. S. Corson. A Lightweight Adaptive Multicast Algorithm. Proceedings of IEEE GLOBECOM, pages 1036-1042, Sydney, Australia, December 1998.

[8] Hui Cheng and Jiannong Cao (2008), The Hong Kong Polytechnic University, "A Design Framework And Taxonomy For Hybrid Routing Protocols In Mobile Ad Hoc Networks", IEEE Communications, Surveys 3rd Quarter 2008, Volume 10, No. 3.

[9] J.J Garcia-Luna-Aceves and E.L. Madruga, "The Core-Assisted Mesh Protocol," IEEE Journal on Selected Areas in Communication, vol. 17, no. 8, August 1999.

[10] S.-J. Lee, M. Gerla, and C.-C. Chiang, "On-Demand Multicast Routing Protocol," in Proceedings of IEEE WCNC'99, September 1999.

[11] Sangman Moh, Chansu Yu, Ben Lee, and Hee Yong Youn, "Energy Efficient and Robust Multicast Protocol for Mobile Ad Hoc Networks", Proceedings of the 2002 Pacific Rim international Symposium on Dependable Computing (December 16 - 18, 2002). Proceedings of IEEE Computer Society, Washington, DC, 145.

[12] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Algorithms for

Energy-Efficient Multicasting in Ad Hoc Wireless Networks," Proc. of Military Communication Conference (MILCOM 1999), Vol. 2, pp. 1414-1418, Nov. 1999.

[13] Hui Cheng and Jiannong Cao, "A Design Framework and Taxonomy For Hybrid Routing Protocols in Mobile Ad Hoc Networks", IEEE Communications, Surveys 3rd Quarter 2008, Volume 10, No. 3.

[14] Pariza Kamboj, A.K.Sharma, "Location Aware Reduced Diffusion Hybrid Routing Algorithm (LARDHR)", accepted for ICETET 09, Nagpur, India.

[15] Y. B. Ko and N. H. Vaida, "Location-aided routing (LAR) in mobile ad hoc networks", presented at the ACM/IEEE MobiCom'98, Oct. 1998.

[16] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, "A distance routing effect algorithm for mobility (DREAM)," presented at the ACM/IEEE MobiCom'98, Oct. 1998.

[17] J. Li, J. Jannotti, D. S. J. D. Couto, D. R. Karger, and R. Morris, "A scalable location service for geographic ad hoc routing," presented at the ACM/IEEE MobiCom, Boston, MA, Aug. 2000.

[18] Yufang Zhu and Thomas Kunz, "MAODV Implementation for NS – 2.26" Communications and Networking in China, 2006. ChinaCom apos;06. First International Conference on Volume, Issue 25-27, pp: 1 – 5.

[19] Pariza Kamboj, A.K.Sharma, "MAODV-PR: A Modified Mobile Ad Hoc distance Vector Routing Protocol with Proactive Route Maintenance", VOYAGER - The Journal of Computer Science & Information Technology, Vol. 6, No. 1, Jan-June 2008, pp. 35-41.

[20] Elizabeth M. Royer and Charles E. Perkins, "Multicast Operation of the Ad-hoc On-Demand Distance Vector Routing Protocol", in Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom'99), Seattle, WA, USA, August 1999, pages 207-218.