A Hybrid Multi Meshed Tree Routing Protocol for Wireless Ad hoc Networks

Scott Pudlewski, Nirmala Shenoy, Yamin Al-Mousa, Yin Pan, John Fischer
Rochester Institute of Technology, Rochester, NY

Abstract

A proactive routing protocol called Multi-Mesh Tree (MMT) was developed for use in wireless ad hoc network to extend connectivity from an Internet gateway to around 20 mobile nodes in a city area. In the work presented here, we extend MMT to wireless ad hoc networks of around one hundred nodes through a clustering algorithm that is integrated into the MMT creation. The proposed scheme uses a hybrid approach, where the proactive MMT is used for intra cluster routing while a reactive MMT (RMMT) introduced in this article is used for inter cluster routing. We further propose a novel route discovery and route recording scheme using route request and route response messages but has low flooding overheads and exhibits high route stability under high node mobility conditions. We apply the proposed RMMT scheme to provide connectivity among moving teams of ground troops and present simulation results based on a study of this scenario.

1. Introduction

In wireless ad hoc networks the communicating nodes form a network independent of a centralized control. Each individual node takes on some of the routing responsibility. When one node needs to send data to another node beyond its transmission range, the intermediate nodes will forward the data along to the destination [1]. However, to decide which nodes are to forward the data a route discovery process is necessary. Based on the route discovery process there are two broad categories of routing protocols – proactive and reactive. Proactive routing protocols form routes to all nodes in the network irrespective of whether or not there is a requirement by a node to send data. These routing protocols exhibit low initial latency when a sender wants to send data to a destination due to available pre-discovered routes. However, proactive routing results in higher overheads due to continual route discovery which consumes bandwidth that could otherwise have been utilized for sending data. The overheads vary based on factors like node mobility and number of nodes in the network. An example of this kind of routing protocol is Optimized Link State Routing [2]. Reactive routing protocols invoke route discovery only when a node has data to send to a destination node. This allows for considerably reduced overheads, but the initial latency from the time a packet is queued at the sender to the time that packet is sent out will be high due to the route discovery process which precedes the packet delivery, unless there is some cached non stale routes to the destination stored at the sender. Popular examples under this category is Dynamic Source Routing (DSR) [3] and Ad hoc On Demand Distance Vector (AODV) routing [4].

The Multi Meshed Tree (MMT) routing [5] algorithm was developed as a proactive routing scheme to support high route robustness with a quick and easy forwarding approach based on virtual IDs (VID), by leveraging the combined features of a tree and a mesh. The scheme was introduced to support several multi hop mobile nodes in a limited area and with limited wireless hops from an Internet connected gateway (which was the root of the meshed tree) to extend IP services to users in a city area. We extend the MMT algorithm to address routing requirements in wireless ad hoc networks comprising over 50 nodes which are all highly mobile, by introducing a reactive component to the already proposed proactive MMT based routing scheme.

We build on the already gained knowledge that hierarchy helps address scalability in large wireless ad hoc networks and hybrid routing with proactive and reactive routing components helps in reducing routing overheads. The functions used for setting up the proactive routes in MMT also form the multi hop cluster with the cluster head (CH) as the root of the meshed tree. We further introduced a reactive MMT (RMMT) for inter cluster routing. Our scheme draws on the knowledge gained by early research work in this area, but has some distinctive advantages, due to the

This material is based upon work supported by the Air Force Office of Scientific Research under Contract No. FA9550-07-C-0004.

MMT based approach. All three schemes, cluster formation, proactive routing and reactive routing in MMT are based on a common meshed tree algorithm. In the section below we highlight related work.

Zone Routing Protocol (ZRP) [6] proposes a general framework that employs hybrid routing. ZRP defines a zone around each node with any proactive routing scheme to be used in the zone and any reactive routing for communications outside the zone. For flooding the route discovery queries outside the zone they use a bordercast resolution protocol to determine the border nodes in a zone. In MMT, knowledge of border nodes at the cluster head is inherent in the scheme.

In MMT for the reactive route discovery and setup we use source routing principles similar to DSR, but the reactive routes are concatenations of the proactive routes in a cluster based on bidirectional links and as the proactive routes are updated dynamically on changes in link conditions the reactive routes will rarely be stale, which is a distinct advantage for MMT based reactive routes. The RMMT route discovery process follows the typical dissemination of ‘route discovery’ messages and stores a ‘recorded route’ for the distant destination node. However MMT route discovery messages are directed to selected border nodes in the overlapping areas of the clusters and the recorded routes maintain only the CH information, which results in heavy reduction on route discovery messages while also reducing route failure probability to the set of intermediate CHs and not all the intermediate nodes that were used for either forwarding the discovery query messages or returning the route response messages. This reactive routing approach is first of its kind to the best of our knowledge.

Figure 1 is reproduced from [5]. In this figure AG1, AG2 and AG3 are Internet connected gateways. Assume for simplicity that the gateways have VIDs 1, 2 and 3. Each gateway announces its VID. The first hop mobile nodes listen to the announcement and send in a request to be connected. The gateways accept the request and allocate VIDs to them. These VIDs are derivatives of the gateway’s VID; namely mobile node A gets VID 11, B gets VID 12 and C gets VID 13 from gateway AG1. The first hop nodes register with the gateway along with the newly acquired VIDs and their uniqueIDs (UID) which could be their MAC or IP address or any other given ID. After registering, the first hop nodes announce their VIDs and acquire second hop children, which subsequently register with the gateway. The branches of the tree grow using the VIDs. For further details on MMT the reader is referred to article [5]. The Opnet simulation models of MMT were validated with analytical models in [5].

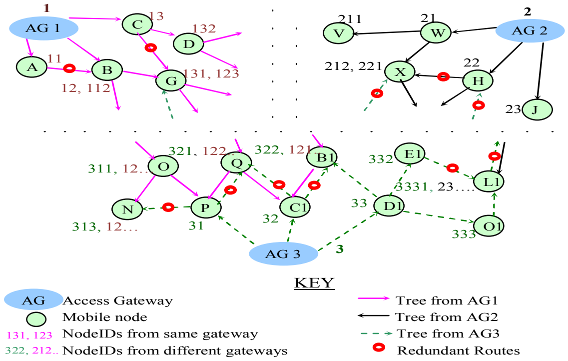

Figure 1 Multi-Meshed Tree in a MANET

2. The hybrid multi -meshed tree

The meshed tree concept explained above was extended to address scalability through clustering. Clusters are formed dynamically and a cluster head (CH) elected based on defined criteria (details of which are not presented here). Without loss of generality we assume a simple heuristic clustering algorithm and once a CH is elected the MMT algorithm is used to create the cluster around the CH. The CH hence, is the root of the meshed tree and multihop client feature is inherent in the proactive route set up. Within the cluster the MMT proactive routing scheme will be used. For inter cluster routing, a novel reactive scheme which extends MMT for route discovery and inter cluster routing/forwarding is introduced in this paper.

As per the work under the Air Force contract (details given in foot note under page 1) MMT was evaluated for robust connectivity among highly mobile unmanned aerial vehicles (UAV) travelling at 200-400 Kmph that collect sensed data from ground sensors at predefined CH nodes, which was successfully tested for connectivity in a network of 100 UAVs. The work was further extended to provide connectivity and communications between teams of ground troops that required RMMT for inter team communications.

In Figure 2, we show a snapshot from our Opnet simulations that capture the MMT creation in a network of around twenty mobile nodes, with a cluster size of 10. The icons are the mobile nodes. The circled nodes with IDs 1003 and 1000 are the CHs. The two trees rooted at these CHs are distinguished by solid and dotted lines. The meshing of the tree branches can be noticed from the multiple VIDs of the nodes shown beside each node. Note the overlapping of the two trees as nodes in the boundary acquire VIDs under the two clusters. Such nodes are crucial for inter cluster connectivity and reactive routing.

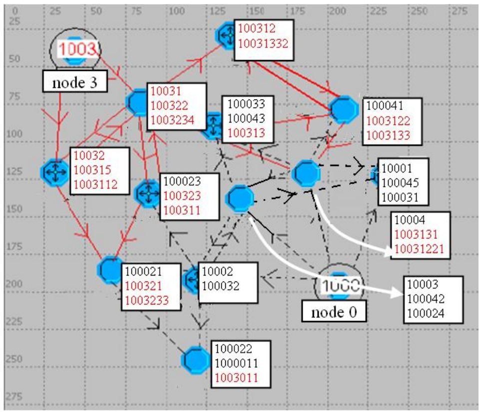

Figure 2 MMT based Cluster Creation

3. The reactive multi meshed tree

Reactive routing is essential when two nodes belonging to different clusters want to communicate. This requires three processes; namely 1) the route discovery from sender node to destination node; 2) the data packet forwarding and lastly 3) inter-cluster route maintenance for the duration of the session.

In MMT the CHs have knowledge of their cluster clients as they maintain cache of their clients UIDs and their multiple VIDs; hence it is only necessary for a route request (or data packet) to reach the CH, to which the destination node belongs to and the CH can then forward the request (or data packet) to this node.

Reactive routing in wireless ad hoc network faces two major challenges 1) route discovery which can introduce significant overhead traffic due to query flooding and 2) route maintenance especially in networks with highly mobile nodes which can introduce several route breaks during a session resulting in repeated route rediscoveries [7]. The situation gets unmanageable when the size of the network and node mobility increases. In RMMT, the route to a destination node is recorded as a set of clusters (CH VIDs) along the path to the destination. This allows for less broken routes; since even when the border and intermediate nodes in the path move out of range, as long as the CH continues in that cluster the packet forwarding to the subsequent cluster enroute to the destination cluster will be successful. The number of route discovery packets is kept low, as in each cluster the route discovery messages are handled by the CHs and one of the boundary nodes between two distinct clusters. These are explained in detail in the sections that follow.

3.1 RREQ packet handling

When a node has data to send to a destination it has to discover a route to the destination. The sending node only has knowledge of the destinations UID and not it's VID. The VID defines the node's location in a cluster in the MMT. In our studies we assumed a UID that is allocated during deployment. Without loss of generality, the cluster head uses this UID to create its VID. For example, a cluster head with UID 7 has a VID that is also 1007.

To find the destination node's VID, the sending node will send a route request packet (RREQ) to each of its cluster heads. When a CH gets the RREQ packet, it will check to see if the destination is one of its clients; if this is true it will forward the RREQ packet to the client and abort its route discovery process. If the node is not a client the CH will create a list of border nodes for neighboring clusters and forward the RREQ to each of these border nodes after inserting itself into the path recorded in the RREQ packet. Unlike typical query flooding used in most route discovery algorithm, in RMMT the RREQ packets are not broadcast but are uni-directed to selected nodes.

When forwarding an RREQ packet to the boundary nodes the CH will identify the best suited border node based on defined criteria (in our case it was the least hops) and forward the RREQ to such border nodes. In the RREQ packet the forwarding CH will also identify the next cluster head that the border node should forward the RREQ packet to in order to avoid looping as the border node could belong to several clusters. The border nodes or the intermediate nodes VIDs are not recorded as part of the route in the RREQ packet. This process where the CH and border node forward is repeated at each cluster. For example, assume a CH with VID 1005 has an RREQ packet to forward, and it has border nodes to clusters 1002, 1010, and 1000, it would include the VID of 1002, 1010, and 1000 in the RREQ packets to each of the border nodes. When CH 1002 gets the RREQ packet and prepares to forward it to its neighboring clusters, it will not forward to clusters 1005, 1002, 1010, or 1000 as it received the RREQ from 1005 and knows that 1005 already sent the RREQ to 1002, 1010 and 1000. This allows for great reduction in the number of RREQ packets transmitted while still covering all possible clusters thereby providing redundancy in paths taken by the RREQ packets.

In Figure 3, we show the path taken by a typical RREQ packet. Node 17 wants to discover a path to Node 6, which happens to be a cluster head. It sends a RREQ packet to its cluster head 1002. Cluster head 1002 identifies node 13, as a boundary node between clusters 1002 and 1001. It forwards the RREQ packet to this node by recording the border node's VID 1002311 as the interim destination of the RREQ packet. At the intermediate nodes 10023 and 100231, the packet is forwarded at the MAC process by using the successive nature of the VIDs without going to the routing process. When border node 1002311 receives the packet it will forward the RREQ packet to the next cluster head identified in the RREQ packet namely cluster head 1001. Cluster head 1001 in turn will forward the RREQ packet to border node 9 on towards cluster 1006. Note that the path of the RREQ goes twice through node 9, which is due to the clustering approach and the fact that traffic has to be forwarded through the cluster heads.

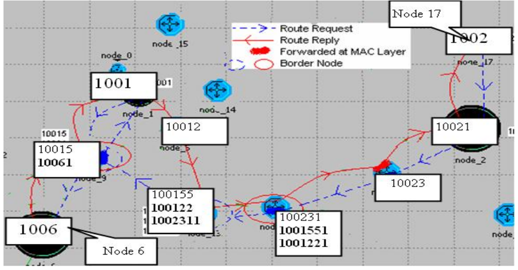

Figure 3 Route Request Reply in RMMT

3.2 RREP packet handling

When a RREQ packet has reached the destination cluster's CH, the CH will forward the RREQ to the destination node. That node will then respond with a route reply (RREP) packet to the CH from which it received the RREQ. The RREP holds the information of the recorded CHs in reverse order to that noted in the received RREQ packet and RREP packet will follow this path of CHs. The border nodes can be different as seen in Figure 3. As the RREP packet travels back to the sender, the intermediate cluster heads will maintain a state indicating the next cluster head that has to forward the data packet addressed to the destination, which is similar to the process adopted by AODV, the difference being that only cluster heads maintain this state.

The above outlined approach leads to highly reduced number of nodes that can affect the route and result in route breaks – i.e. only a change in cluster heads can break a route. The routing table entry at the sending and receiving nodes is the next cluster head VID and the distance recorded in number of cluster-hops. When the RREP reaches the original sending node, the node will wait a predefined amount of time before it begins to send data. The predefined wait time is to ensure the return of multiple RREP messages.

4. Data forwarding and redirection

We present a typical data forwarding mechanism using MMT. This forwarding is done at the MAC process without intervention of the routing process. From the work presented so far we note that the packets are forwarded from a cluster client to the cluster head or vice versa. Assume a packet is destined to a cluster client with VID of 1007362, which means that once a packet reaches the cluster head 1007 it could be delivered to the cluster client via the path 1007 → 10073 → 100736 → 1007362, where 10073 is the first level parent, 100736 is the second level parent and so on.

When a data packet reaches a CH enroute, the CH will forward to the current best suited border node between itself and the next cluster towards the destination. This forwarding algorithm has unique advantages for routing in large scale ad hoc networks as it allows for maximum flexibility in the face of high node mobility. There will however be cases in which the path will break and data will be blocked at an intermediate node where the node will resort to either redirection or route rediscovery; redirection in MMT is explained next.

When data packet is blocked at a node that is not a cluster head, the node can direct the data to the cluster head using one of its other VIDs under the same cluster. For example a source node with VID 1003421 forwards a data packet to the cluster head 1003 to be subsequently forwarded to a destination in another cluster. The packet is forwarded at the MAC process to node 100342 towards the cluster head. Assume that at this node because of its movement it has lost VID 100342, but has another VID 100321 in the same cluster. After the retries, this intermediate node will then use VID 100321 to forward the data packet to cluster head 1003.

It can also happen that node with VID 100342, lost that VID due to its movement and has no other VIDs in cluster 1003, but has a VID under cluster 1002 namely 100231. It will then forward the packet to the cluster head 1002, which may have a recorded route for that destination and is able to forward the packet.

If a cluster head finds that the first recorded route towards the destination is not available, it can use one of the other routes available for the same destination. Thus there are several recovery options. In the worst case a cluster head may store the data packet for some time, reinitiate the route discovery process and then forward the packet.

The above processes would seem similar to route salvage adopted by several reactive routing algorithms, but the routes within a cluster in MMT are dynamically updated and proactively maintained and hence probability of the stored routes being stale is very low.

5. Opnet simulations

The RMMT was tested for wireless ad hoc networks with sizes of 20 and 50 mobile nodes. Two mobility models were used. Random waypoint mobility model which is commonly associated with pedestrian type movement; and circular trajectories to stress test the route robustness of the routing algorithm. In the random waypoint model the pause time was set to 0. The circular orbits had a radius of 10 km, and the transmission range was maintained at 15 km. These values would introduce several route breaks as the nodes try to get better routes, while keeping the nodes from getting completely out of range from one another. For the random waypoint model, there were 50 nodes uniformly distributed in a 15 km by 15 km space with a 3 km transmission range. Two files sizes were used - 10 Kbytes and 50 Kbytes. Packet sizes were varied from 1 Kbyte, 2 Kbyte and 5 Kbytes. The number of data-sending nodes was varied from 3, 5 to 10 nodes. The corresponding receiver nodes were selected to be at the other end of the network. At the physical layer data rates were maintained at 33 Mbps, which is a typical X band data rate used in UAV communications and also would be the combined data rate if all three channels defined for 802.11 MAC were collapsed and hence suitable for communications among ground troops. All physical layer parameters including interference and collisions were modeled using Opnet provided models. As these were the preliminary studies of the proposed algorithm terrain models were not used. Packets with a single bit error were dropped and no forward error correction used.

Several stress tests were conducted to determine the limits when the performance dropped below certain specified values. Several simulations with various seeds were run and the performance metrics averaged over all simulation runs. We have not provided comparison with other hybrid approaches, as there is no study available for those approaches under the given stress conditions. The performance metrics targeted were success rate, end to end packet delivery latency and total file delivery latency.

Success rate is defined as ratio of the number of packets received correctly at the destination node to the number of packets sent to the destination as a percentage.

End to end packet latency is the duration from the time a packet was sent by the source to the time it was received at the destination to determine the delay experienced in the network as imposed by the routing algorithm.

File transfer delay included the time from when the file was sent by the sender to the time the file was either completely received at the receiver or till there were no packets belonging to that file in the network.

6. Performance analysis

We restrict the performance presentation in this section to the 50 mobile node network scenario with circular orbits. The results are presented as three sets. The first set presented under section 6.1, shows recorded metrics with normal pedestrian and slow vehicle speeds of 1 m/s to 5 m/s. In table 1 below

σ = packet size in Kbytes,

ρ = the success rate in percentage and

δ = end to end packet latency in milliseconds.

6.1 The 50 node scenario

Table 1 Statistics for 50 node scenario

File Size		3 sending nodes		5 sending nodes		10 sending nodes	
	σ	ρ	δ	ρ	δ	ρ	δ
10 Kb	1	100.00	114.86	100.00	177.84	100.00	216.21
	2	100.00	146.86	100.00	210.35	100.00	443.34
	5	100.00	357.46	100.00	450.79	100.00	539.77
50 Kb	1	100.00	555.74	100.00	830.63	97.64	888.93
	2	100.00	553.60	100.00	792.54	99.96	1073.46
	5	100.00	559.58	100.00	848.07	100.00	1292.59

As can be seen in the table above with the nodes sending a file of size 10 Kbytes, we get a 100% success rate, whatever the packet size and irrespective of whether 3, 5 or 10 nodes were sending the files. The average latency increases with the packet size as the probability of collisions increases (the MAC process was a modified 802.11 MAC to handle bulk and burst data transfer to make the best use of the transient routes). When sending 50 Kbytes file, the per packet average latency increases and there is a drop in the success rate with increasing number of sender nodes. The latency is varying because of the redirection and rediscovery employed by MMT and RMMT respectively.

6.2 Stress test scenarios

In this section, we present the test scenarios, where we stressed the RMMT algorithm by increasing the traffic, by introducing higher node mobility and by changing the movement trajectory to introduce higher route breaks. For the stress testing we focused only on the 50 node scenario with both the random waypoint model and the circular trajectories. There are 3 nodes sending data to receivers selected at the other end of the network. The files size was maintained at 10 Kbyte – packet size 1 Kbyte. There are 3 cases presented here based on the type of stress test conducted. The first stress test introduced a continuous stream of data. While presenting the result for this test we also present the impact of the redirection capability of MMT and the rediscovery capability of RMMT.

High Mobility – Random Waypoint: we used the random waypoint mobility model and varied the speeds as indicated in Table 2. Pause time was set to zero. The range in the top row indicates that in one scenario we had nodes with speeds ranging from 10 to 30 meters per second as shown in the first column or from 70 to 100 meters per second as shown in the last column. We conducted the tests to note when the success rate would drop to 60%. The recorded results are self explanatory.

Table 2 Random Waypoint Mobility Model

Speed in m/s	10 - 30	30 - 50	50 - 70	70 - 100
Success Rate %	94.22	81.17	71.82	60.37
Packet Latency (ms)	737.99	1333.49	1807.72	1979.57
File Transfer Time (ms)	728.36	1328.05	1820.50	1964.93

Circular Trajectories: Table 3 records the data collected with circular orbits. As stated earlier, this scenario would result in several route breaks. The success rate starts dropping and latency increases as the speeds increase. In some cases more dropped packets result in reduced latency as can be noted while comparing the entries in column 1 and 2 and 1 and 3. A success rate of greater than 80% can be noticed. The packet latency variations are due to the redirection and rediscovery, which can change considerably depending on the number of packets that were to be redirected or for which paths had to be rediscovered.

Table 3 Circular Trajectories

Speed in m/s	10	15	20	25	30	40
Success rate %	95.45	97.20	93.43	94.28	82.40	84.10
Packet Latency (ms)	422.15	436.51	494.13	385.20	1243.89	1236.01
File Transfer Time (ms)	462.27	444.71	539.83	413.89	1260.60	1265.68

7. Conclusion

In this paper we have extended a proactive routing algorithm that leverages the advantages of a mesh and a tree called the Multi Meshed Tree (MMT) protocol, which was used for a limited number of nodes to address scalability by introducing clustering and its reactive counterpart the Reactive MMT or RMMT. This protocol is shown to provide highly robust links across multiple hops at high levels of mobility. We also describe how RMMT is able to greatly reduce the number of flooding packets needed to discover multiple, reliable routes. The highlights of RMMT are: 1) highly reduced flooding for route discovery and; 2) higher stability of discovered routes in the face of node mobility. The performance metrics focused on were success rate, packet delivery latency and file delivery latency. Given the critical applications to which these algorithms had to be applied we stress tested the proposed scheme to identify the limits when the performance drops below a specified level.

8. References

[1] Z. J. Haas, M. Gerla, D. B. Johnson, C. E. Perkins, M. B. Pursley, M. Steenstrup, C. K. Toh, and J. F. Hayes, "Guest editorial wireless ad hoc networks," *Selected Areas in Communications, IEEE Journal on,* vol. 17, pp. 1329-1332, 1999.

[2] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," in *Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International*, 2001, pp. 62-68.

[3] D. Johnson, Y. Hu, D. Maltz *The Dynamic Source Routing Protocol (DSR)*, IETF RFC 4728, February 2007; http://www.ietf.org/rfc/rfc4728.txt

[4] C. Perkins, E. Belding-Royer, S. Das, *Ad-hoc on-Demand Distance Vector (AODV) Routing*, IETF RFC 3561, July 2003; http://www.ietf.org/rfc/rfc3561.txt

[5] N. Shenoy and P. Yin, "Multi-meshed tree routing for Internet MANETs," in *Wireless Communication Systems, 2005. 2nd International Symposium on*, 2005, pp. 145-149.

[6] Z. J. Haas and M. R. Pearlman, "The performance of a new routing protocol for the reconfigurable," in *Communications, 1998. ICC 98. Conference Record.1998 IEEE International Conference on*. vol. 1 Atlanta, GA, USA, 1998, pp. 156-160.

[7] C. Jiwei, L. Yeng-Zhong, Z. He, G. Mario, and S. Yantai, "Robust Ad Hoc Routing for Lossy Wireless Environment," in *Military Communications Conference, 2006. MILCOM 2006*, 2006, pp. 1-7.