Journal of Theoretical and Applied Information Technology

www.jatit.org

DESIGN OF A QUALITY OF SERVICE-BASED MULTICAST ROUTING PROTOCOL FOR MOBILE AD HOC NETWORKS

NATARAJAN MEGHANATHAN

Assistant Professor, Department of Computer Science, Jackson State University, Jackson, MS 39217, USA

E-mail: natarajan.meghanathan@jsums.edu

ABSTRACT

We propose a Quality-of-Service (QoS)-based multicast routing protocol, referred to as QoS-MR, for mobile ad hoc networks (MANETs). QoS-MR is a receiver-initiated mesh-based multicast routing protocol and can work for both combinatorially stable and unstable networks. We explain in detail the procedures for mesh construction (initiation by the source, propagation of the control packets by the intermediate nodes and route selection by the receiver) and mesh maintenance/repair. We take into account QoS metrics such as bandwidth, end-to-end delay and link lifetime. The source specifies the desired QoS characteristics of the application in the control messages broadcast for mesh formation and the intermediate nodes forward the control messages only if they could satisfy the QoS demands and reserve the resources requested by the application. The receiver selects the route that best satisfies the QoS constraints and notifies the source. A congregate of all these source-receiver routes leads to the formation of a multicast mesh that also involves the links between the forwarding nodes of these routes. Mesh/route repair is conducted through an expanding ring search (ERS) approach to minimize the control overhead. An alternate route that best satisfies the QoS constraints is incorporated into the mesh.

Keywords: *Quality of Service, Multicast Routing Protocol, Mobile Ad hoc Networks, Link Expiration Time, Bandwidth, End-to-end Delay, Mesh*

1. INTRODUCTION

Quality of Service (QoS) to the user is a guarantee, agreement or commitment by the network to provide a set of measurable pre-specified service attributes such as trans-network delay, delay variance (jitter), available bandwidth, probability of packet loss, etc [1]. In order to honor the guarantee, enough resources must be available in the network during service invocation [1]. The QoS policy should provide pre-emptive priority for control packets than user-level data packets. Also, a QoS policy should allow varying priorities among user-level data flows. Two main tasks constitute QoS routing: (1) To find a suitable *route* between the source and the destination(s) that has the necessary resources available to meet the QoS constraints for the desired service and (2) Reserve resources along the chosen *route*.

Although a lot of work has been done to provide QoS in the wired Internet, they cannot be directly applied to mobile ad hoc networks (MANETs) because of the latter's dynamically changing topology and bandwidth constraints [2]. Wireless links have characteristics (bandwidth, error rate) that change with time. Enough radio channel capacity must be available in order to ensure an end-to-end delay upper bound as part of the QoS guarantees. It will be very difficult to find a loop-free path if the network topology changes before the last topology updates are propagated to all the pertinent nodes. An ad hoc network is said to be *combinatorially stable* if and only if the topology changes occur sufficiently slowly to allow successful propagation of all topology updates as necessary [1]. An ad hoc network is said to be *QoS-robust* only if all the specific set of QoS guarantees are maintained regardless of the topology updates that may occur within the network [1]. An ad hoc network is said to be *QoS-preserving* if the QoS guarantees can be maintained during the interval between the end of a successful topology update and the occurrence of the next topology change event [1]. By this definition, a *QoS-robust* network is *QoS-preserving* but the converse need not be true [1].

QoS routing depends heavily on the current network state (local state and global state). The local state information is maintained at each node and it includes the queuing delay, node residual capacity, propagation delay, bandwidth and cost metric for each of the node's outgoing links. The global state of the network is constructed by exchanging the local state information among nodes at appropriate moments and hence represents the totality of the local state information of all nodes in the network. The global state information cannot be completely true since topology updates through out the network cannot happen instantaneously [1]. For ad hoc networks with high mobility, it is almost infeasible to obtain an accurate global state of the network [1]. A hierarchically organized clustering network has been proposed in [3] to provide QoS support in ad hoc networks. The entire network is partitioned into clusters in a hierarchical fashion and aggregated partial global state information is maintained only at the cluster level.

The widely used Integrated Services/ Resource Reservation Protocol (IntServ/RSVP) [4] technique in wired networks cannot be applied for MANETs because of the following constraints in MANETs [6]:

(1) *State explosion:* The amount of state information to be stored at intermediate nodes increases proportionally with the number of flows. This is a problem in the current wired networks too. A huge storage and processing overhead is incurred for the mobile hosts whose computing and storage resources are scarce.

(2) *Channel contention:* The RSVP signaling packets contend with the data packets for channel access and consume a substantial amount of bandwidth in the bandwidth-constrained MANETs.

(3) *Packet classification:* Every mobile host should act as a router performing the QoS functions of classification, admission control and scheduling, etc. This is a huge burden on the resource-limited hosts.

Differentiated services (DiffServ) [5] model is proposed for wired networks to handle the scalability problem of IntServ/RSVP. DiffServ defines the DS field, which is a layout of the type of service (TOS) bits in the IP header, and a base set of packet forwarding rules called the per-hop behavior (PHB). The boundary routers at the boundary of the network use classification, marking, policing and shaping mechanisms to control the entering traffic. Interior routers just forward the traffic by offering the PHB associated with the marked DS field. Since DiffServ is lightweight in the interior routers, it may be a possible solution for providing QoS in MANETs [2]. But, there are still some challenges faced. Intuitively the source node can act as the boundary router and the other nodes along the forwarding paths from the sources to destinations act as interior routers. But in a MANET environment, the source node cannot be predefined and so all nodes need to possess the capability of interior and boundary routers. Service Level Agreements (SLAs), indispensable part of the DiffServ model in wired networks, are a kind of contract between the customer and his Internet Services Provider (ISP) specifying the forwarding services the customer's traffic should get from the ISP's network [6]. But, since there is no obvious scheme for mobile nodes to negotiate traffic rules, it would be tough to make SLAs in MANETs.

QoS signaling can be either in-band (control information carried along with data packets) or out-of-band (explicit control packets) [6]. RSVP is an out-of-band signaling system whose use in bandwidth-constrained MANETs will impose a significant signaling overhead. Also, control packets contend with the data packets for transmission channel access. Since control packets are piggybacked with data packets, there is no channel contention in an in-band signaling system. An absolute in-band signaling system (all control information piggybacked with data packets) is also not desirable since in the case of a unidirectional data flow from the source to the destination, a feedback control message back to the source is not feasible. In the context of a bandwidth and power constrained MANET environment, designing a simple and lightweight signaling system is more important than a powerful but complex signaling system.

The rest of the paper is organized as follows: In Section 2, we discuss some of the current proposals for QoS support in MANETs. Section 3 discusses the design of our proposed protocol. Section 4 concludes the paper. Throughout the paper, the terms 'packet' and 'message' are used interchangeably. They mean the same.

2. CURRENT MANET QOS SUPPORT

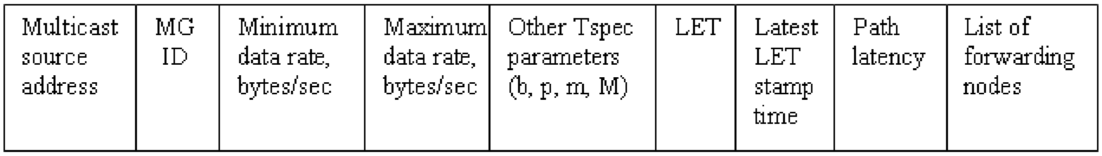
A Flexible QoS Model for MANET (FQMM) has been proposed in [7], that combines the advantages of per-flow granularity in IntServ and service differentiation in DiffServ. Similar to DiffServ, FQMM defines three types of nodes: ingress, interior and egress nodes. The sender of the data is the ingress node. Interior nodes forward the data for other nodes. The destination node is the egress node. But, the role of a node will adaptively change because of the dynamically changing topology. The provisioning scheme (used to determine and allocate resources at various nodes) in FQMM [7] is a hybrid of the per-flow provisioning in IntServ and the per-class provisioning in DiffServ. Per-flow granularity is preserved for a small portion of the MANET since a large amount of traffic belongs to per-aggregate flows (per-class granularity) [2]. Traffic conditioners, responsible for marking and re-marking of traffic streams, are placed at the ingress nodes.

INSIGNIA [8], the first signaling protocol designed solely for MANETs, carries control signal information in the IP option of every IP data packet. The packet classification module classifies the incoming packets and forwards them to the appropriate modules like the routing, INSIGNIA, local applications and packet scheduling modules [8]. If an INSIGNIA option is included in the incoming IP packet, it is forwarded to and processed at the INSIGNIA module using a weighted-round-robin scheduling scheme that takes location-dependent channel access conditions into account. INSIGNIA uses a soft state approach to manage flow state information and is specifically designed fast flow reservation, restoration and adaptation algorithms for adaptive real-time services in MANETs [8]. If the resource requirement can be satisfied and the admission control module lets the flow in, INSIGNIA allocates bandwidth to the flow. If the required resource is unavailable, the flow will be offered the best-effort service. Rejection and error messages are not sent if the resource requirements cannot be satisfied. INSIGNIA sends QoS reports to the source informing it of the status of the real-time flows, changes in the network topology and end-to-end QoS conditions. Based on the feedback information, the source takes appropriate actions to adapt the flows to the network conditions. The drawback with INSIGNIA is the need to maintain flow state information in the mobile hosts, which may present a scalability problem when deployed for moderate and large-scale MANETs.

Two routing techniques are presented in [1] for combinatorially stable QoS-preserving networks: one based on only the availability of local state information and the other including the mostly inaccurate global state information. A probe packet with the appropriate node identity and QoS information is used to identify a feasible route with the desired QoS characteristics. QoS routing with only local state information can be based on two different distributed routing algorithms [1]: source-initiated routing and destination-initiated routing. Destination-initiated routing mitigates the penalties of flooding of the probe packets by intermediate routers in source-initiated routing. A ticket-based probing technique is used to identify a feasible route in routing techniques based on the imprecise knowledge of the global states. Probes are sent across links whose QoS characteristics are slowly varying in time. The number of tickets carried by the probes decreases with the feasibility of finding a QoS route. Probes are used to store information regarding multiple feasible routes rather than the intermediate routers. For QoS-preserving, QoS-routing in MANETs, once the broken routes are detected, they are either repaired or the flow rerouted on an alternate route with the desired QoS. The beaconing protocol for detecting adjacent neighbors is used to detect a broken route. When imprecise routing information is used for the QoS route between a source-destination pair, periodic refresher packets are sent by the destination back to the source. The QoS route is declared unavailable and associated resources released if the packets fail to arrive with in a predetermined time interval.

In [1], multiple redundant routing mechanisms are also considered to minimize the effect of route failures. A simplest and highly redundant technique is to establish for a single flow, multiple alternate routes with the same QoS guarantee and use them simultaneously. The alternate routes may be preferably disjoint with duplicate packets discarded at the destination. An intermediate level of redundancy technique would be to have the routes and associated resources reserved and rank-ordered but not used until the primary route fails. At the lowest level of redundancy, resources may not be reserved and only the alternate routes identified. Alternate paths are checked for availability of resources when the primary path fails.

A bandwidth-constrained QoS routing algorithm based on the distance vector protocol has been proposed in [9], but fails to accommodate the effects of imprecise network state information. A highly sophisticated technique for controlling QoS in large ad hoc networks has been proposed in [3]. It uses the concepts of multi-layered adaptive control in hierarchically structured multi-cluster organizations. More details regarding the cluster dynamics, mobility management, resource reservation, route repair and router movement on QoS can be found in [3]. Two new QoS routing techniques using link-state protocols as the underlying mechanism are proposed in [10] to reduce the routing update overhead. While one technique selectively adjusts the frequency of routing table updates, the other reduces the size of the update messages using a hierarchical addressing approach.



Multicast source address	MG ID	Minimum data rate, bytes/sec	Maximum data rate, bytes/sec	Other Tspec parameters (b, p, m, M)	LET	Latest LET stamp time	Path latency	List of forwarding nodes

Figure 1: BRQ-TSPEC Message

3. DESIGN OF THE QOS-MR PROTOCOL

QoS in ad hoc multicast routing is currently an open research issue that has been addressed only by a few researches. A concept called "Dynamic QoS" has been proposed in [11], where a resource reservation request specifies a range of values and the network commits to provide service at any specific point within the range [11]. Applications request for QoS by specifying their minimum acceptable and maximum utilizable service levels. Even though the concept of "Dynamic QoS" claims to support multicasting in MANETs, details of how the resources are reserved in a multicast flow, how link failures are recovered, etc. are not addressed in [11]. We make use of their concept of "range of service levels" and extend it to multicast. We propose a protocol for QoS in multicast routing for ad hoc networks based on receiver-initiated mesh-based multicasting. The protocol is robust and independent of whether the ad hoc network is combinatorially stable or not. We assume the following: All the nodes run a GPS clock such that the time in all the clocks is synchronized. Unusual cases of a node's power going down or the node suddenly moving far away from the rest of the network are not considered.

3.1. BRQ-TSPEC Message Initiation

The source node advertises the multicast group id, the source address and the Tspec of the application packets it is going to multicast by sending an initial broadcast message called BRQ-TSPEC to all the nodes in the network. Tspec in RSVP defines the token bucket traffic specification parameters (r,b,p,m,M), where r is the average rate, b is the token bucket depth, p is the peak rate, m is the minimum policed unit and M is the maximum packet size [6]. One Tspec for one application is used per BRQ-TSPEC packet but the source can send Tspecs for multiple applications in separate BRQ-TSPEC packets distinguishing the applications using a global application id (similar to the DS field in DiffServ). A minimum and maximum acceptable level of service range (also called data rate) for the application packets and the maximum end-to-end delay tolerable for the application are specified in the BRQ-TSPEC message. Also, included in the BRQ-TSPEC message is a field for the link expiration time (LET), an empty field for the transmission path latency (which will be updated with the summation of individual link latencies) and an empty forwarding nodes field. Figure 1 shows the structure of a BRQ-TSPEC message. It can be seen that our approach combines the features of a multicast initialization message with the features of the PATH message of RSVP. By doing this, we are avoiding the channel contention that would be created when individual control packets are sent using RSVP.

3.2. Prediction of the Link Expiration Time

Given the motion parameters of two neighboring nodes, the duration of time the two nodes will remain neighbors can be predicted as follows: Let two nodes *i* and *j* be within the transmission range of each other. Let (x_i, y_i) and (x_j, y_j) be the co-ordinates of the mobile hosts *i* and *j* respectively. Let v_i, v_j be the velocities and Θ_i, Θ_j, where $(0 \le \Theta_i, \Theta_j < 2\pi)$ indicate the direction of motion of nodes *i* and *j* respectively. The amount of time the two nodes *i* and *j* will stay connected, D_{i-j}, can be predicted using the following equation:

$$D_{i-j} = \frac{-(ab+cd)+\sqrt{(a^2+c^2)r^2-(ad-bc)^2}}{a^2+c^2}$$

where,
$a = v_i \cos\Theta_i - v_j \cos\Theta_j$; $b = x_i - x_j$; $c = v_i \sin\Theta_i - v_j \sin\Theta_j$; $d = y_i - y_j$

3.3. BRQ-TSPEC Message Propagation

All the nodes in the network maintain a multicast routing table (MRT). Nodes receiving the BRQ-TSPEC message immediately record or update their MRT with the multicast group id (MGID), the source address, the upstream node to reach the source, minimum and maximum data rate acceptable to the application, the current LET value in the broadcast message, the GPS time at which this value is stamped in the packet and the current path latency in the broadcast message. This is done irrespective of whether the nodes forward the BRQ_TSPEC message or not. This is to ensure that all the nodes in the network are aware of the MGID, the multicast source and the application characteristics so that in future, if they would like to become a member, they can send a JOIN_REQUEST using these characteristics. A multicast group (MG) flag is also included in the MRT. The MG flag will be activated when the node becomes a member of the forwarding mesh.

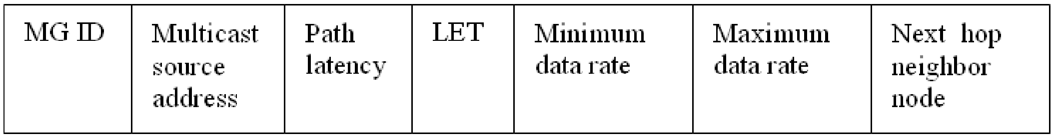
Neighboring nodes that receive the BREQ-TSPEC from the source determine the LET of the link to the source and update the LET field in the BREQ-TSPEC message with the new value and the GPS clock time at which it is determined. The node then determines the bandwidth level of the incoming link through which the BRQ-TSPEC message came. If the bandwidth level of the incoming link is less than or equal to the minimum level in the Tspec of the BRQ-TSPEC message, then the message is discarded at the node and is not forwarded further. Instead the intermediate node sends a BRQ-ADVERTISEMENT (ADVT) message to its neighboring and downstream nodes. The BRQ-ADVT message includes all the fields in the received BRQ-TSPEC message to the downstream nodes, but the nodes that receive the BRQ-ADVT could not process it or use it for making QoS reservations. The BRQ-ADVT packet is sent so that all the downstream nodes can update their MRT with the MGID, the multicast source address and the characteristics of the application packets sent by the source. In future, if the downstream node wishes to join the group, it can send an Expanding Ring Search, ERS-REQUEST packet to join the group. This is explained in detail in Section 3.5. If the bandwidth of the incoming link is greater than the minimum level in the Tspec, the maximum level in the Tspec is then updated with the measured bandwidth level of the incoming link. The transmission path latency field is then updated with the value of MTU/incoming-link bandwidth plus the scheduling delay at the node. The node also appends its address in the forwarding node field of the BRQ-TSPEC message. The message is then rebroadcast to neighboring nodes.

Intermediate nodes receiving the BRQ-TSPEC message, subtract the GPS clock time at which the LET is set in the message from their current GPS clock time. This difference is then subtracted from the LET value set in BRQ-TSPEC and the current GPS clock time at the node is updated in the GPS clock time field. The node then determines the LET of the incoming link. If the LET of the incoming link is more than the current LET in the BRQ-TSPEC message, then the current LET is left unmodified. Otherwise, the LET field in the BRQ-TSPEC message is updated with the new LET value and the GPS clock time set to the current clock time. This sort of fine granular procedure is needed, since we are proposing this approach to work for even a non-combinatorially stable network. Other processes such as updating the transmission latency, determining the incoming link bandwidth and updating the maximum data rate field, etc are the same as explained in the previous paragraph.

3.4. Route Selection by the Receiver

BRQ-TSPEC messages that can successfully pass through the intermediate nodes (the intermediate nodes commit to offer service above the minimum acceptable level of the application) reach the receiver nodes. Receiver nodes that are interested in the multicast session, update the entries into their MRT similar to the other nodes. Also, the receiver nodes maintain a multicast session table in which they store the MGID, the multicast source address, the path latency, LET, minimum and maximum levels of data rate and the next hop neighbor node information. All these information are obtained from the BRQ-TSPEC messages that reach the receiver. If the sum of the path latency and the end-to-end queuing delay of the BRQ-TSPEC message is less than the end-to-end delay bound, the route specified in the broadcast message is accepted and the other fields in the broadcast message are entered to the multicast session table. Otherwise, the route is considered not useful and the broadcast message is discarded. The structure of the proposed multicast session table at a receiver is shown in Figure 2.

The receiver nodes have two choices in choosing the route to reach the sender. They can choose the most stable route to the multicast source by selecting the route that has the highest LET and an acceptable end-to-end delay. The other alternative is to select the route with the least end-to-end delay and use the LET of that route irrespective of the latter's value. The receiver sends a JOIN-RSPEC message back to the source, specifying the minimum and maximum data rate of the selected route, the LET of the selected route and the rest of the Tspec parameters. The forwarding path for the JOIN-RSPEC message is copied from the BRQ-TSPEC message, which is kept in a small-size buffer temporarily before being discarded.



MG ID	Multicast source address	Path latency	LET	Minimum data rate	Maximum data rate	Next hop neighbor node

Figure 2: Multicast Session Table

3.5. JOIN-RSPEC Message

An intermediate node upon receiving the JOIN-RSPEC message, checks whether it is in the next hop field. If the node finds itself in the next hop ID field, it then determines whether it can offer the maximum level of data rate specified in the JOIN-RSPEC message. If the bandwidth of the outgoing link to the next upstream node is less than the minimum data rate in the JOIN-RSPEC message, then the message is discarded and an error message RESV-FAIL is sent to the receiver. Otherwise, if the bandwidth of the outgoing link is greater than the minimum level but less than the maximum level of data rate, the maximum level of data rate is updated to the bandwidth of the outgoing link to the upstream node. If the bandwidth of the link to the upstream node is greater than or equal to the maximum level of data rate in the JOIN-RSPEC message, the current maximum level of data rate is retained. The intermediate node then updates the LET in the JOIN-RSPEC by comparing with the LET of the incoming link and updating the LET in the JOIN-RSPEC message as explained before for the BRQ-TSPEC message. In either of the latter two cases of link bandwidth comparison and if the LET value is greater than zero, the intermediate node identifies itself as a member of the multicast forwarding mesh and activates the MG flag for the corresponding MGID in its MRT. The LET values are updated per hop both in forward and reverse transmission because we want the protocol to work for even non-combinatorially stable networks. The LET value present in both the BRQ-TSPEC message and the JOIN-TSPEC message will be the LET value of the critical link in the routes of these two packets. This critical link will be the first link along the route to break down. The current maximum level of data rate for the multicast session is updated in the MRT. The JOIN-RSPEC message is then rebroadcast to the neighbor nodes. The above process is repeated at all the intermediate forwarding nodes until the message reaches the source node.

A JOIN-RSPEC message reaching the source node is considered to be successful because it can reach the source only after the network confirms the reservation of resources. Otherwise, the reservation request would have been dropped at an intermediate node and an error message sent back to the receiver. A forwarding mesh based on receiver-initiated requests is then formed and the source starts sending the actual data packets.

3.6. Maintenance of the QoS-MR Mesh

Since all the nodes in the network are synchronized using the GPS clock, the receiver node can predict the failure of the critical link in the selected route. The receiver node continuously decreases the LET value (using the GPS clock) of the selected route in its multicast session table. When the LET value of the critical link approaches zero, the receiver initiates an expanded ring search (ERS) for an alternate stable route to the source. Note that The LET of the critical link may not have actually decreased to zero. It might have also remained constant or increased. The receiver node is just reducing the LET of the critical link only in its own multicast session table in order to be fault-tolerant and have an alternate route readily available when the current route fails.

The ERS-REQUEST packet has the following fields: the MGID, the multicast source address and the current minimum and maximum data rate. The scope of the ERS-REQUEST packet is increased gradually upon failure to find a route. When a non-member node receives a non-duplicate ERS-REQUEST packet, it just forwards the packet to its neighboring nodes. Non-duplicate broadcast packets in our proposed protocol are detected using the conventional sequence number based approach. When a member node or a forwarding node of the multicast group receives the ERS-REQUEST packet, it checks whether the LET value in its MRT for the multicast group is valid. The difference between the current GPS clock time and the clock time at which the LET is stored in the MRT is determined. A new LET value is then computed by subtracting the difference in the clock times from the current LET value. If this new LET value approaches zero, or is less than or equal to zero, the member node or the forwarding group node sends a scoped ERS-REQUEST packet for itself to its upstream nodes. Here, we also propose an alternative approach. The forwarding group node similar to the receiver node can also constantly decrease the LET values in their MRT based on the GPS clock time and initiate an ERS search. Even though this approach would incur a slight processing overhead, it would guarantee a high percentage of QoS to the flow.

If the member node or the forwarding node determines that it is still the active member of the multicast group (determined after validating its LET), it then checks the minimum and maximum data rate in the ERS-REQUEST packet with that in its MRT. A JOIN-ACK packet is then sent back to the receiver node confirming resource reservation and route availability. If the maximum data rate in the MRT is less than that in the ERS-REQUEST, the maximum data rate in the JOIN-ACK is set to the maximum data rate in the MRT of the acknowledging node. Otherwise, the maximum data rate in the ERS-REQUEST packet is copied to that in the JOIN-ACK packet. The receiver node may get JOIN-ACKs from more than one member node or forwarding group node. In such a scenario, the receiver node selects the path with the least delay or the highest LET as explained before. The receiver node stores all the received paths in its MRT and the selected route in its multicast session table. It then sends a JOIN-CONF to the node that sent the corresponding selected route. The node then starts receiving the packets from that node through the selected route. This mechanism of local QoS recovery decreases the risk of QoS violation upon link failure.

We use a soft-state approach by which the sender refreshes the multicast entries in the forwarding nodes by sending periodic BRQ-TSPEC packets. The sender could also advertise the change in the application characteristics through these periodic BRQ-TSPEC messages. The local QoS recovery approach as explained above helps to have sufficiently large BRQ-TSPEC update intervals and reduce the overhead of flooding. The receiver node may also initiate an ERS-REQUEST packet if it fails to get data packets from its upstream node for consecutive fixed time intervals.

Our approach can be easily adapted to tree-based stability-oriented multicast routing protocols like the Associativity-based Ad hoc Multicast (ABAM) routing protocol [12]. In the ABAM protocol, the broadcast message does not include the application characteristics and is flooded irrespective of the number of associativity ticks. Associativity ticks are beacon-like messages periodically broadcast by a node to its neighbors. The receiver does not continuously monitor for path failure in ABAM. Intermediate nodes do not provide any QoS guarantees to flows. Our QoS-MR protocol can be adapted and used over ABAM for providing QoS in tree-based multicasting.

4. CONCLUSIONS AND FUTURE WORK

The high-level contribution of this paper is the design of a new QoS-based multicast routing protocol for MANETs. In addition, we also reviewed the QoS features in traditional wired networks and explored their application in MANETs. We discussed the major challenges that are faced during the design of QoS in MANETs. The proposed QoS-MR protocol has been designed to work for both combinatorially stable and unstable networks. QoS-MR can also be easily adapted to provide QoS in tree-based protocols such as ABAM that uses associativity ticks to determine the stability of a route. QoS-MR is the first such approach to accommodate the three different metrics such as the link expiration time, link bandwidth and end-to-end delay in its multicast mesh creation and maintenance procedures. In the near future, we will be working on the implementation of the proposed QoS-MR protocol and its comparison to the other QoS-based MANET multicast routing protocols.

REFRENCES:

[1] S. Chakrabarthi and A. Mishra, "QoS Issues in Ad hoc Wireless Networks," *IEEE Communications Magazine*, vol. 39, no.2, pp.142-148, Feb. 2001.

[2] K. Wu and J. Harms, "QoS Support in Mobile Ad hoc Networks," *Crossing Boundaries – An interdisciplinary Journal*, vol.1, no.1, pp.92-104, Fall 2001.

[3] R. Ramanathan and S. Steenstrup, "Hierarchically Organized, Multi-hop Mobile Wireless Networks for Quality of Service Support," *Mobile Networks and Applications*, vol.3, no.1, pp.101-119, June 1998.

[4] R. Braden, D. Clark and S. Shenker, "Integrated Services in the Internet Architecture – an Overview," *IETF RFC 1663*, 1994.

[5] S. Blake, "An Architecture for Differentiated Services," *IETF RFC 2475*, 1998.

[6] P. P. White: "RSVP and Integrated Services in the Internet: A Tutorial," *IEEE Communications*, vol. 35, no. 5, pp.101-106, May 1997.

[7] H. Xiao, W. K. G. Seah, A. Lo and K. C. Chua, "A Flexible Quality of Service Model for Mobile Ad hoc Networks," *Proceedings of the 51st IEEE Vehicular Technology Conference*, pp. 445 – 449, May 2000.

[8] S. B. Lee, and A. T. Campbell, "INSIGNIA: In-band Signaling Support for QoS in Mobile Ad hoc Networks," *Proceedings of the 5th International Workshop on Mobile Multimedia Communications* (MoMuC'98), Berlin, Germany, October 1998.

[9] T. Chen, M. Gerla and J. T. Tasi, "QoS Routing Performance in a Multi-hop Wireless Network," *Proceedings of the IEEE International Conference on Universal Personal Communications*, pp. 557 – 561, 1997.

[10] A. Iwata, C. C. Chiang, G. Pei, M. Gerla and T. W. Chen, "Scalable Routing Strategies for Ad hoc Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol.17, no.8, pp.1369-1379, August 1999.

[11] M. Mirhakkak, N. Schult and D. Thomson, "Dynamic Quality of Service for Mobile Ad hoc Networks," *Proceedings of the 1st ACM International Symposium on Mobile Ad hoc Networking and* Computing, pp. 137 – 138, August 2000.

[12] C. K. Toh, G. Guichal and S. Bunchua, "ABAM: On-demand Associativity-based Multicast Routing for Ad hoc Mobile Networks," *Proceedings of the 52nd IEEE VTS Fall Vehicular Technology Conference*, vol.3, pp.987-993, 2000.