

© 2010 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Multicast QoS Routing using Collaborative Path Exploration

Jaipal Singh

Digital Ecosystem and Business Intelligent Institute
Curtin University of Technology
Perth, Australia
e-mail: j.singh@cbs.curtin.edu.au

Abstract—Quality of Service (QoS) is one of the most active research areas in networking. The most fundamental requirement for QoS routing is the ability to find and maintain a network path that provides the required network resources between two or more nodes. In this paper, we present a distributed collaborative multicast QoS routing architecture that uses a semi-greedy probing heuristic to quickly find a QoS path between a joining node and the multicast tree. The proposed architecture will enable the routers along the path to intelligently and dynamically discover a QoS path. Any router that receives a probe will only know its neighbours and it will create a link to the previous router from where the probe comes from. The proposed architecture is a tree-initiated QoS search and the first QoS packet to reach the joining node will be used as the QoS path. Analysis of this method shows that the path search time and message overhead is lower than other similar schemes.

Keywords—quality of service (QoS); multicast; computer networks

I. INTRODUCTION

A lot of research has been done in multicast networks in the 1990s and early 2000. Multicast was developed by Steve Deering [1] as a more efficient method to transmit IP packets to multiple receivers. While it was a very promising technology for group communication, it was not widely adopted due to security issues, service pricing and limited bandwidth availability. In the past few years, there has been a dramatic increase in network bandwidth that has led to ubiquitous connectivity to the Internet. This has led some service providers (ISP) to look at using multicast technology, particularly for high bandwidth applications like IPTV.

With the increase in Internet use, there is more competition to use the available network bandwidth. Therefore, quality of service (QoS) is becoming very important as more and more real-time multimedia content is sent through the network. QoS is the ability for a network to provide the resources required for selected network traffic to reach their destination successfully without causing other best-effort network traffic to fail. The aim of QoS multicast routing is to find a path that meets the QoS requirement from a sender to one or more receivers on the network. The most commonly used QoS performance parameters are latency, delay variation (jitter), throughput and packet loss rate [2].

Previous researchers have developed QoS routing heuristics for finding an optimal QoS path between a joining node and the multicast tree in dense-mode and sparse-mode source based trees (SBT) networks. Other researchers typically use probes to find a QoS path in a multicast shared tree network. Refer to [3-5] for details on these heuristics and QoS probes.

In this paper, we are proposing a distributed collaborative multicast QoS routing architecture that uses semi-greedy probes to find a QoS path that explores the network from the sparse mode multicast tree (SBT or ShT) to the joining node. This work has been previously published in [6] and this paper will provide more details of how the routers will process the QoS exploration probes and provide more simulation comparisons of our proposed architecture against other similar Multicast QoS architectures. We will show that our multicast QoS architecture can find a successful QoS path quicker than the previous QoS probe schemes and with less probe packets sent through the network.

The remainder of the paper is organized as follows: a brief overview of previous work by other researchers into multicast QoS probes is presented in section II. In Section III, we describe our collaborative multicast QoS probe architecture. Section IV will provide simulation results of our proposed QoS probe architecture along with comparison with earlier works. Section V concludes this paper.

II. RELATED WORK

Current routing protocols use the shortest path between two nodes as the routing metric and have to be extended to support QoS routing metrics. Based on these metrics and the network state information, a QoS routing protocol can be used to find one or more feasible QoS paths.

The QoS path is a path bounded by one or more constraints. However, finding a path that satisfies multiple QoS constraints might not be solvable in polynomial time. Chen and Nahrstedt lists the combinations of multiple QoS constraints that can and cannot be solved in polynomial time [7].

In addition to finding a feasible QoS path, a QoS routing mechanism also needs to establish a QoS path based on the paths identified and to maintain the path for the duration of the QoS session [8].

In this paper, we will only look at QoS routing for finding a feasible QoS path. We will not look at mechanisms to establish and maintain the QoS path. Due to space limitations, this paper will limit itself to finding the least delay constrained QoS path.

Delay or latency is the time a packet takes to travel from the source node to the receiver nodes on the multicast tree. In a multicast tree, delay is a tree-constrained routing problem where the packet travel time delay from a source to the receiver(s) in a tree must not exceed a specified value. Real-time interactive applications like video conferencing and voice over IP (VoIP) cannot tolerate long delays otherwise the conversation between the sender and receiver becomes impractical. This is similar to continuous media broadcasts like IPTV.

The current assumption is that the network delay of a link is proportional to the distance of the link. We also assume that the input and output queue delay will be calculated as part of the total incoming or outgoing delay. With these assumptions, the shortest path between two nodes will have the least delay between the said nodes. Probing based QoS routing algorithms do not assume the shortest path has the least delay and are generally used for finding a delay constrained QoS path in a sparse-mode multicast network.

Probe based QoS routing algorithms will probe the network link to find a feasible path from a source node to a multicast tree by using a single path algorithm or multiple path algorithm. Single path algorithms will search for only one path that meets the QoS requirements. If the path fails to meet the QoS requirement, the routing algorithm will start a new search for another path. Multiple path algorithms will try to find several paths that meet the QoS requirements concurrently. The multicast member will select the ‘best’ path from a list of available paths.

WAVE [9], YAM [10] and QoSMIC [11-12] are QoS routing algorithms that use multiple path routing while QMRP [13] uses single path routing and only switches to multiple path routing if the single path routing fails to find a QoS path.

The spanning joins (YAM) architecture finds one or more candidate paths from a joining node to the multicast tree. The joining node uses broadcast with reverse path forwarding (RPF) to send out join messages on all links towards the multicast tree. Once an on-tree router receives the broadcast packet, it will probe for the QoS requirement on the reverse path taken by the probe packet. The joining node will evaluate which path contains the best QoS from the reverse probes it receives and joins the tree using the best path found.

QoSMIC probes for a QoS path in two directions, from the source node and from the multicast tree. It improves on YAM by limiting the broadcast search area to limit resource overheads and introduces the notion of a QoS manager. If the multicast tree is beyond the probe search area, the manager node can be used to choose candidate routers to initiate the probe from the multicast tree to the source node. This method is only effective if the joining node knows about the

manager node’s address. Like YAM, the joining node will select the best QoS path from all the received probes to join the multicast tree.

The QoS-aware multicast routing (QMRP) protocol uses both single path and multiple path routing to find a QoS path. Unlike YAM and QoSMIC, QMRP starts probing for a QoS path from the joining node to the on-tree router on the shortest path between the two nodes. If the single path fails the QoS probe check, the probe will go back one router and switch to multiple path probing on every link except for the link leading to the failed path and the origin of the probe. Once the probe successfully reaches an on-tree router, an ACK will be sent on the reverse path of the probe back to the joining node where it will select the best path (if there are more than one ACK message).

QMRP does not flood the network with as many packets as YAM or QoSMIC to find a path to the multicast tree. However, in the worst case, QMRP might find itself backtracking all the way to the source node to perform a multi-path search if it cannot find a QoS path at later routers. Unlike YAM and QoSMIC, QMRP is only interested in finding a successful QoS path rather than the optimal QoS path from the joining node to the multicast tree.

The WAVE routing algorithm finds a QoS path from the multicast tree to the joining node. The joining node will inform the tree that it wants a QoS path and the tree will probe the network to find a QoS path to the joining node. WAVE was developed only for source-based routing multicast trees.

In a large network, both YAM and QoSMIC might take a long time to converge and have high message complexity. The joining router has no way of knowing how long it will take for every probe to reach itself. The time a probe is sent depends on the location of the tree reached by the spanning join. The RPF used by both these protocols assume that the links are symmetrical. If the network has asymmetrical links, the convergence time might be much longer than if the network was symmetrical. The WAVE protocol also suffers from this convergence problem since nodes nearer the source node will send a probe before the farther nodes in the tree. The collaborative Multicast QoS routing protocol described in this paper will improve on these previous QoS protocols.

III. PROPOSED COLLABORATIVE MULTICAST QoS ROUTING ARCHITECTURE

In the following, we denote a network topology $G = (V, E)$ with a multicast shared tree, $T = (V', E')$ where $V' \subseteq V$ and $E' \subseteq E$ with each edge $e = uv \geq 0$. If $u \in (V - V')$ is a joining node's connecting router, our problem is to find a path from u to an on-tree node $v' \in T$ that satisfies the required delay constrained QoS using our proposed Explore Best Path Message (EBPM) architecture.

The EBPM architecture minimises the use of multiple path QoS routing which leads to high broadcast overheads in large networks. Instead of delaying the reservation of QoS by exploring for the optimal QoS path, we will only explore for the optimal path if the shortest path cannot provide the

requested QoS. Therefore, we use single path probe on the shortest path between the joining router and the multicast tree and only use multiple path probing if the single path fails to provide the required delay constrained QoS.

Even though EBPM can be used to find a QoS path for any type of sparse mode multicast tree (SBT or ShT), there have been many proposal for QoS routing in SBT multicast networks. In this paper, we will show how EBPM is used with a bi-directional multicast shared tree like the new PIM-BIDIR protocol [14].

A. Single Path QoS Exploration

If we look at Fig. 1, router u wants to join the multicast tree and sends a QoS probe within the PIM-BIDIR Join-Request message to the core router ($V'1$) of the multicast tree, T which consists of on-tree routers $V'1$, $V'2$ and $V'3$. The join message will use the underlying unicast routing table to reach the core router on the shortest path from u to $V'1$.

As the join message traverses the network, the QoS delay information from each router along the path will be accumulated and stored in the QoS probe packet. This operation does not use much overhead compared to the regular PIM-BIDIR Join-Request message since the router does not make any QoS decisions during this initial stage. The main purpose of this initial Join-Request is to inform an on-tree router that a member is interested in joining the multicast tree rather than to find a QoS satisfying path to the tree.

This Join-Request message will reach an on-tree router v'_n on the shortest path using the underlying unicast routing table. The example in Fig. 1 shows that the shortest path between the joining router to the multicast tree is $u \rightarrow A \rightarrow V'1$. The on-tree router $V'1$ will perform a QoS eligibility test. If the Join-Request passes the eligibility test, the on-tree router will send a Join-Ack message back to the joining router on the reverse path used by the Join-Request message.

B. Multiple Path QoS Exploration

The EBPM multipath QoS routing is only used if the

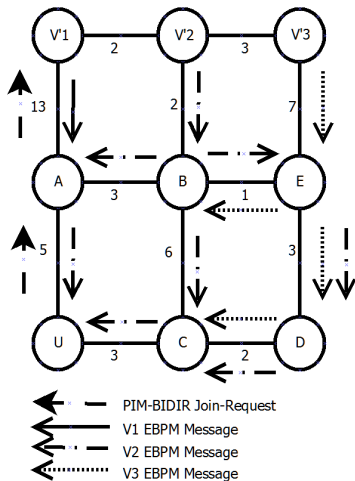


Figure 1. The joining node's router u explores QoS path to PIM-BIDIR multicast tree.

Join-Request message fails the QoS eligibility test. If we assume that the QoS needed to join the tree is 15 ms, the diagram shows that the shortest route path used by the Join-Request does not satisfy the required QoS. The on-tree router $V'1$ will multicast an Explore Best Path Message (EBPM) initiation message to all other on-tree routers. Every on-tree router that receives this initiation message will broadcast an EBPM to every qualified outgoing neighbour which are not on-tree routers.

A non-tree node v_2 is a qualified neighbour for some node v_1 if the link v_1v_2 meets the required QoS delay constraint. Each EBPM message is uniquely identified by each of the on-tree router v'_n (source) that initialises the message, the joining router u (destination), the multicast group address and a sequence number. The sequence number is incremented whenever a source issues a new EBPM to the same destination for the same group after the expiry of the previous EBPM.

Unlike the Join-Request message which only collects QoS information to be processed by an on-tree router, the QoS value in an EBPM message will be evaluated by each router that receives it. The router will create a routing state for the received EBPM message and then compare the collected QoS value with the required QoS value. If the QoS value is within the QoS bound, the router will then update the EBPM QoS value, replicate the packet and forward it on every qualified outgoing link not directly attached to an on-tree router or on the reverse path of the EBPM.

Every qualified neighbour router v_n that receives an EBPM will process the packet according to these six conditions as detailed below. These conditions are illustrated in Fig. 2.

- a) Condition 1 - If v_n has no qualified neighbours, then the EBPM will be dropped by v_n . Router A in Fig. 1 will drop the EBPM from on-tree router $V'1$ since it cannot find a path with the required delay QoS.
- b) Condition 2 - If the EBPM time-to-live (TTL) expires, the EBPM will be dropped.
- c) Condition 3 - If a previously received EBPM has a better path QoS than the current EBPM, the current EBPM will be dropped. In the example in Fig. 1, router B will drop the EBPM message from router E since it already knows a better QoS path to the on-tree router ($V'2$).
- d) Condition 4 - If the current EBPM has the same QoS update value as the QoS value in the routing table, the router will use this new path as a backup path. The router will create a backup reverse link to the previous router that sent this EBPM and drop the EBPM. In Fig. 1, the EBPM received by router C from router D ($V'2 \rightarrow B \rightarrow E \rightarrow D$) has the same QoS as the EBPM from router B ($V'2 \rightarrow B$). Router C will use router D as a backup link in case the link to router B fails.
- e) Condition 5 - If router v_n receives an EBPM for the first time and it is within the required QoS value, router v_n will create a reverse link to the last router that sent this

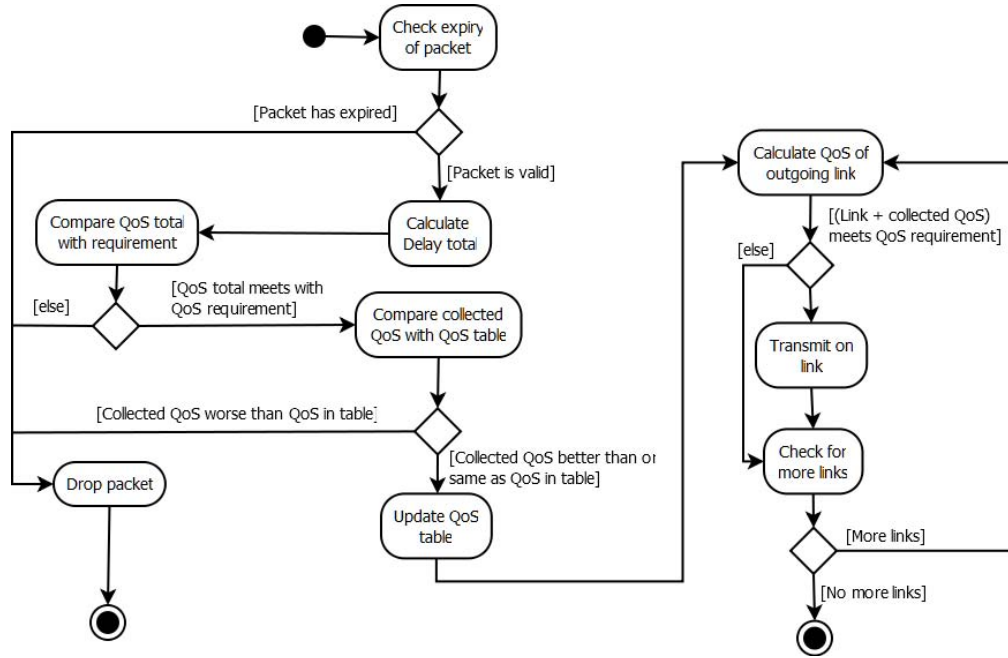


Figure 2. EBPM activity diagram at a router.

EBPM and rebroadcast the EBPM to all the qualified neighbours.

f) Condition 6 - If this is not the first EBPM received by a router and the QoS value is better than the QoS in the reverse link table, the router will change the old reverse link to the new router which sent the better EBPM packet. The router will then forward the EBPM packet on all qualified outgoing links. To keep the router's routing table small, the router will only keep a routing state to the last router that sent the best EBPM packet.

The way a path QoS is calculated depends on whether the QoS measurement is for the outgoing link, incoming link or both links. The outgoing link follows the direction from the on-tree router to the joining router and vice-versa for incoming link. Normally multicast receivers require the QoS on the outgoing link while multicast senders require QoS on the incoming link.

Since all routers know their outgoing link QoS, the router will replicate the EBPM and update the EBPM QoS of the link it goes out on if only the outgoing link QoS is measured. In the case of the incoming link QoS, a router will send the EBPM to the next hop router which will fill in the QoS value of the link it came on (previous router's incoming link). If QoS is required for both incoming and outgoing links, then the routers will perform both operations, i.e. update QoS for downstream link and send the EBPM to a qualified router which will update the QoS for the upstream link. This process will continue until the EBPM process terminates either by finding a successful path or until the packet is dropped.

IV. RESULTS

In this section, we present our simulation results. We compared the message complexity (number of messages exchanged until the algorithm terminates) and the convergence time of our algorithm with YAM [10], QoS MIC [11-12] and QMRP [13].

We used the BRITe topology generator [15] to construct an autonomous system network topology. We generated several random Euclidean graphs for 50 nodes using the Waxman method to represent an autonomous system. The nodes are connected by asymmetrical links and uses a Heavy-Tailed distribution. The graphs represent a hierarchical architecture of a metropolitan area network, offering high redundancy at every layer in the hierarchy. Thus the number of edges in these graphs are more than $O(n)$.

Once the network topology is generated, we selected three random connected nodes to be the multicast tree on-tree routers. The joining node was randomly selected from this topology. The joining node will join the multicast tree on a QoS path that meets the delay constrained requirements of 150ms.

In our simulations, we simulated the base YAM protocol, QoS MIC using both the spanning join and manager assisted probing methods with a radius limit of 2, QMRP and EBPM. For EBPM, we simulated that all on-tree routers sent out the EBPM message at the same time.

Table 1 summarises the total number of QoS exploration probe messages sent through the network to find a delay constrained QoS path.

TABLE I. QoS PROBE MESSAGE COMPLEXITY FOR SEARCHING DELAY CONSTRAINED PATH

Protocol	Number of Probe Messages	
	Joining node less than 3 hops away from Multicast Tree	Joining node more than 3 hops from Multicast Tree
YAM	182	230
QoSMIC	29	54
QMRP	6	85
EBPM	6	41

When the joining node is close to the multicast tree, both QMRP and EBPM use the same number of messages to find the QoS path. This is because they use the QoS path on the shortest path between the joining node and multicast tree. While YAM and QoSMIC have higher messages, both of them found the optimal path in the network. The best case for QMRP provides similar results as EBPM.

We show the worst case scenario for the QoS routing algorithms when the joining node is further away from the multicast tree. In our simulations, the shortest path does not provide the required QoS and the network has very limited QoS paths as the distance increases. While all the message complexities increase, EBPM is lower than all the other algorithms. We show that the worst case for QMRP can drastically increase the message complexity as the algorithm starts to backtrack and initiate multiple path routing. In this simulation, there is no QoS path that satisfies the delay requirements.

Fig. 3 shows the time taken to find a QoS path for all the QoS algorithms. This graph shows the amount of time taken for EBPM, QMRP and QoSMIC to complete their exploration of the network for a QoS path. It does not show the completion of the YAM algorithm. As mentioned earlier, both EBPM and QMRP find a QoS path faster than all the other algorithms as they use single path routing on the shortest path. While this does not provide the optimal QoS path, it finds a QoS path 3 times faster than QoSMIC.

Fig. 4 shows the time taken to find a QoS path between a joining node that is more than 3 hops away from the multicast tree. It only shows a small timeframe of the QoS

probe exploration as the complete graph will make it difficult to compare the different QoS routing algorithms as YAM and QMRP values are go outside the graph boundaries. The simulation shows that EBPM completes the exploration of the network faster than the other QoS routing algorithms. As the network cannot provide the QoS path, QMRP performs many backtrack operations with multiple path exploration. As can be seen from the figure, QMRP takes a long time to explore all paths and performs worst than QoSMIC or EBPM.

From our simulation results, we can derive the following conclusions:

a) In QoSMIC, if a joining node is r hops away from the multicast tree, the spanning join radius limit should be at least r hops for it to find a path an on-tree router. If the radius is less than r hops, QoSMIC cannot find a path to the tree without relying on the tree manager. Since the tree manager only probes on the shortest path, there is no guarantee that the shortest path can provide the required QoS.

b) If the topology is dense, YAM will find a lot of paths to the on-tree routers although most of these paths do not provide the required QoS. In the case of QoSMIC, as the spanning join radius increases, the number of paths found will similarly increase. However, like YAM, the majority of these paths will not provide the required QoS and lower the percentage of successful paths found. EBPM will always provide 0% or 100% since there are always only two conditions – an EBPM will reach the joining router or not.

c) The EBPM message complexity is lower than the other QoS routing protocols because whenever a router receives an EBPM exploration packet that is worse than any previous EBPM packet or when a router has no qualified neighbours, the router will discard the EBPM exploration packet.

V. CONCLUSION

This paper describes the EBPM architecture, a collaborative Multicast QoS Routing architecture, which uses both single path routing and multiple path routing to find a

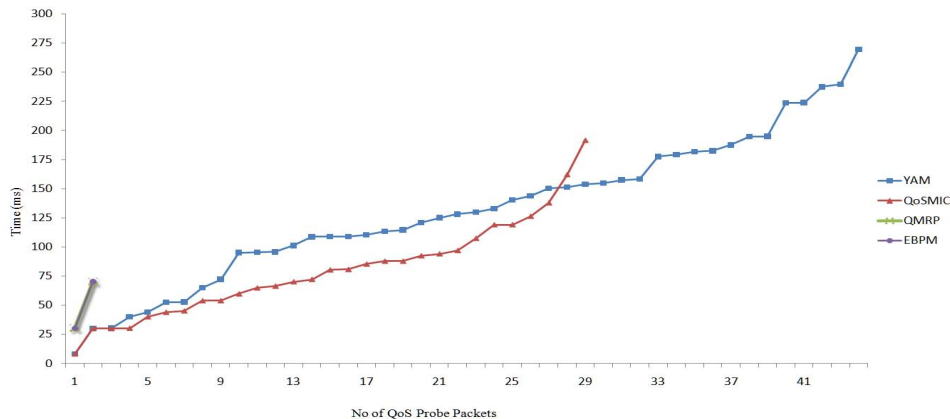


Fig. 3. Time taken to find delay constrained QoS path when joining node is close to multicast tree.

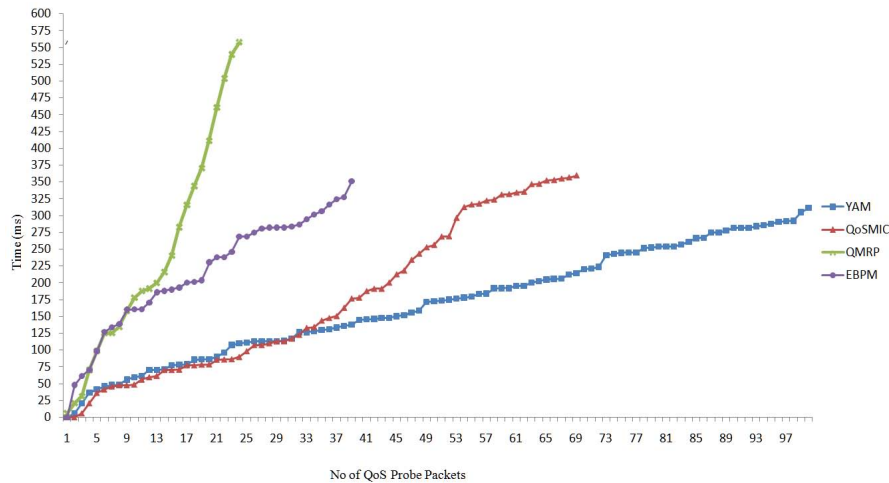


Fig. 4. Time taken to find delay constrained QoS path when joining node is far from multicast tree.

delay constrained QoS path between a source node and multicast tree. In order to reduce the QoS path exploration overheads, the EBPM architecture first tries to find a QoS path on the shortest path between the joining node and the multicast tree. If the shortest path does not provide the required QoS, the on-tree routers will use multiple path routing to search the network for a path with the required QoS.

We have shown through simulations, that the multiple path exploration phase of EBPM will find the optimal delay constrained QoS path faster than YAM and QoSMIC algorithms. For multicast trees close to the joining node, QMRP and EBPM provide similar results. If the multicast tree is further away from the joining node, EBPM can find the optimal QoS path quicker than the other QoS protocols and with lower message complexity. This is due to the collaborative nature of EBPM, where the routers can dynamically change the QoS path links to provide the best QoS path between the joining node and multicast tree. We will extend this work to explore bandwidth constrained QoS paths in future papers as well as study the feasibility of using EBPM in wireless ad hoc networks.

REFERENCES

[1] S. E. Deering and D. R. Cheriton, "Multicast routing in Datagram Internetworks and Extended LANs," *ACM Transactions on Computer Systems (TOCS)*, vol. 8, no. 2, pp. 85-110, 1990.

[2] J. Shin, D. C. Lee and C. C. J. Kuo, *Quality of Service for Internet Multimedia*: Prentice Hall PTR, 2003.

[3] B. Wang and J. C. Hou, "Multicast Routing and its QoS Extension: Problems, Algorithms and Protocols," *IEEE Network*, vol. 14, no. 1, pp. 22-36, 2000.

[4] P. Paul and S. V. Raghavan, "Survey of QoS Routing," in *Proceedings of the 15th International Conference on Computer Communication*, pp. 50-75.

[5] P. Paul and S. V. Raghavan, "Survey of Multicast Routing Algorithms and Protocols," in *Proceedings of the 15th International Conference on Computer Communication*.

[6] J. Singh, P. Veeraraghavan and S. Singh, "QoS Multicast Routing using Explore Best Path," *Elsevier Computer Communications*, vol. 29, no. 15, pp. 2881-2894, 2006.

[7] S. Chen and K. Nahrstedt, "An Overview of Quality of Service Routing for Next-Generation High-Speed Networks: Problems and Solutions," *IEEE Network*, vol. 12, no. 6, pp. 64-79, 1998.

[8] G. Apostolopoulos, *et al.*, "QoS Routing Mechanisms and OSPF Extensions," 1999.

[9] E. Biersack and J. r. Nonnenmacher, "WAVE: A New Multicast Routing Algorithm for Static and Dynamic Multicast Groups," in *Proceedings of the 5th International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, pp. 228-239.

[10] K. Carlberg and J. Crowcroft, "Building Shared Trees Using a One-to-Many Joining Mechanism," *ACM SIGCOMM Computer Communication Review*, vol. 27, no. 1, pp. 5-11, 1997.

[11] M. Faloutsos, A. Banerjee and R. Pankaj, "QoSMIC: Quality of Service Sensitive Multicast Internet Protocol," in *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 144-153.

[12] S. Yan, M. Faloutsos and A. Banerjee, "QoS-aware multicast routing for the Internet: the design and evaluation of QoS MIC," *IEEE/ACM Transactions on Networking (TON)*, vol. 10, no. 1, pp. 54-66, 2002.

[13] S. Chen, K. Nahrstedt and Y. Shavitt, "A QoS-aware multicast routing protocol," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 12, pp. 2580-2592, 2000.

[14] M. Handley, I. Kouvelas, T. Speakman and L. Vicisano, "Bi-directional Protocol Independent Multicast (BIDIR-PIM)," IETF, Internet Draft draft-ietf-pim-bidir-0, 22 October 2005.

[15] A. Medina, A. Lakhina, I. Matta and J. Byers, "Boston University Representative {Internet} Topology Generator (BRITE)," ed. Available from: <http://www.cs.bu.edu/brite/> [Accessed 12 December 2009].