Multicast routing for ad hoc networks with overload prevention for group members

Bür, Kaan; Ersoy, Cem

2004

Document Version:
Peer reviewed version (aka post-print)

Link to publication

Citation for published version (APA):
Bür, K., & Ersoy, C. (2004). *Multicast routing for ad hoc networks with overload prevention for group members*. Poster session presented at ISTMS - the 13th IST Mobile and Wireless Communications Summit, Lyon, France.

Total number of authors:
2

Multicast Routing for Ad Hoc Networks with Overload Prevention for Group Members

Kaan Bür, Cem Ersoy

NETLAB, Bogazici University, Department of Computer Engineering, 34342 Bebek, Istanbul, Turkey
{burk, ersoy}@boun.edu.tr

ABSTRACT

Mobility, quality of service (QoS) support, and multicast routing capability are important requirements for today's networks. This is a result of the changes in wireless users' expectations, which favour group-oriented, high-quality, multimedia communication. Ad hoc networks enhanced with QoS multicasting strategies can provide users with these features. This paper defines the building blocks of an ad hoc QoS multicasting (AQM) protocol, which tracks the availability of resources within a node's neighbourhood based on reservations made previously, and announces it at session initiation. When nodes join a session with certain QoS requirements, this information is updated and used to select the most appropriate routes. AQM is compared to a non-QoS scheme with particular emphasis on preventing overload on multicast group members. It achieves better results through the use of its QoS-related routing decisions, which suggests that QoS is both essential for and applicable to multicast applications in ad hoc networks.

Index terms: Wireless communication, mobile multimedia, ad hoc networks, quality of service, multicast routing.

I. INTRODUCTION

Ad hoc networks are communication groups formed by wireless mobile hosts without any centralised control or established infrastructure. They are becoming increasingly popular as a result of the rapid penetration of mobile telephony and developments in multimedia communication over Internet. While current network systems are primarily designed for one specific type of application such as speech, video or data, the next generation will integrate various functions and applications. Therefore, it is essential that wireless and multimedia be brought together [1].

Ad hoc networks are considered for many group-oriented services and applications, including in-home networking, nomadic computing, wireless local area networks, and communication for disaster relief, public events, and temporary offices. In this regard, they have to support multimedia applications. Thus, quality of service (QoS) becomes a critical issue. QoS support for multimedia applications is closely related to resource allocation, which is to decide how to reserve resources such that QoS requirements of all applications are satisfied [2]. It is a significant challenge to provide reliable high-speed end-to-end communications in ad hoc networks, due to their dynamic topology, distributed control, and multihop connections [3]. Multicasting is a promising technique to utilise resources effectively and provide a subset of network nodes with the service they demand while not jeopardizing the bandwidth requirements of others. The advantage of multicasting is that packets are multiplexed only when it is necessary to reach two or more receivers on disjoint paths. Due to their broadcasting capability, ad hoc networks are inherently ready for multicasting.

It is not an easy task to incorporate QoS to ad hoc multicasting. Incremental changes on existing schemes cannot efficiently address the issues mentioned above. In this paper, the ad hoc QoS multicasting (AQM) protocol is presented to increase multicasting efficiency through distributed QoS management. AQM tracks the availability of resources within a node's neighbourhood based on previous reservations, and announces it at session initiation. During the join process, this information is updated and used to select routes which can satisfy the QoS requirements of the session. Thus, AQM decreases the number of overloaded members significantly. The rest of this paper is organised as follows. Previous research related to ad hoc multicasting is summarised in Chapter II. AQM is introduced in Chapter III. The performance of the proposed system is evaluated in Chapter IV. Concluding remarks and future work are presented in Chapter V.

II. AD HOC MULTICASTING OVERVIEW

There are various protocols developed to maintain a multicast graph and perform routing in ad hoc networks, some of which are summarised below. However, they do not address the QoS aspect of ad hoc communication, which is becoming increasingly important as the demand for mobile multimedia increases.

The on-demand multicast routing protocol (ODMRP) introduces the concept of a forwarding group [4, 5]. Sources periodically broadcast join query messages to invite new members and refresh existing membership information. When a node receives a join query, it stores the upstream node address in its routing table and rebroadcasts the packet. When a node decides to join a session, it broadcasts a join reply. When a node receives a join reply and sees that it is on the path to the source, it sets its forwarding group flag and broadcasts its own join reply after updating the table of next nodes. Periodic join requests initiated by the source have to be answered by session members with join replies to stay in the group.

Multicast ad hoc on demand distance vector (MAODV) routing protocol is derived from AODV [6, 7]. The multicast group leader maintains a group sequence number and broadcasts it periodically to keep fresh the routing information. A node wishing to join a multicast group generates a route request. Only the leader or members of the multicast group may respond to a join request by unicasting a route reply back to the requester, which selects the best route from several replies and enables it by unicasting a multicast activation message to its next hop. Intermediate non-member nodes receiving activation messages unicast them upstream along the best route according to the replies they received previously. Nodes wishing to leave the group unicast a multicast activation message to their next hop with the prune flag set.

Bandwidth-efficient multicast routing (BEMR) finds the nearest forwarding group member for joining nodes [8]. When a new node broadcasts a join request, each node receiving it adds its node ID to the message and increments the hop count before flooding it. Forwarding nodes receive some of these requests, choose the best hop alternative and send a reply packet along the selected path. The requester eventually receives multiple replies, chooses the best hop alternative and sends a reserve packet along the same path. All nodes on this path become forwarding nodes.

Differential destination multicast (DDM) lets source nodes manage group membership, and stores forwarding state information encoded in headers of data packets to achieve stateless multicasting [9, 10]. Join messages are unicast to the source, which tests admission requirements, adds the requester to its member list, and acknowledges it as a receiver. The source refreshes its member list to purge stale members. It sets a poll flag in data packets to force its active receivers to resend join messages. Leave messages are also unicast to the source. Forwarding is based on destinations encoded in headers. Each node checks the header for any DDM block or poll flag intended for it.

Neighbour-supporting multicast protocol (NSMP) utilises node locality to reduce routing maintenance overhead [11]. A new source broadcasts a flooding request. Receivers send replies to their upstream nodes. To maintain mesh connectivity, the source employs local route discoveries by periodically sending local requests, which are only relayed to mesh nodes and their immediate neighbours to limit flooding. Replies are sent back to the source. Nodes farther away from the source have to flood member requests.

Associativity-based ad hoc multicast (ABAM) builds a source-based multicast tree [12]. Association stability based on the number of consecutive beacons received from a neighbour helps the nodes select routes which last longer. Receivers run a route selection algorithm to reply with routes of highest association stability. Upon receiving the replies, the source runs a tree selection algorithm to build the multicast tree, and sends back a setup message. To join a multicast tree, a node broadcasts a request, collects replies from group members, selects the best route with the route selection algorithm, and sends a confirmation. To leave, a notification is propagated upstream along the tree until a branching node or a receiver is reached.

III. The AD HOC QoS MULTICASTING PROTOCOL

The motivation behind QoS support for multicasting in ad hoc networks is that mobile multimedia applications are becoming very popular in group communications. For an efficient QoS multicasting strategy, implementation of QoS classes, bounded delay, and bandwidth reservation are very important. In the following sections, the structure of AQM is defined, which addresses these issues.

A. Quality Classes and Levels

Different QoS classes are necessary to support various types of applications in an efficient manner. In this work, 4 QoS classes are suggested to represent a sample set of applications. 2 quality levels are supported in each class to provide nodes with communication options of less bandwidth. Recent multimedia compression techniques such as MPEG-4 for mobile define methods to encode multimedia data in separate layers, allowing intermediate nodes of a session to extract and send just the base layer and drop the rest without first consulting the originator of the data or decoding and encoding back the whole packet if necessary [13]. Defining QoS levels also limits the amount of information to be transmitted. Otherwise, it is impossible to forward a best QoS combination without making some assumptions or losing some valuable data. It is preferable that nodes inform others on the availability of QoS levels and send updates only when the levels change.

B. Session Initiation and Destruction

A session can be started by any node (MCN_INIT), which broadcasts a session initiation packet (SES_INIT) with a session identity number and an application type. A table of active sessions (TBL_SESSION) is maintained at each node to keep this session definition. Using their session tables, nodes forward initiation packets of new sessions. A membership table (TBL_MEMBER) is used to denote the status of the predecessors (MCN_PRED) which have informed the node on the existence of a particular session, and the QoS level of the path from the session initiator up to that node via that predecessor. The session initiation packet is forwarded as long as the QoS requirements are met. Before the packet is rebroadcast, each node updates its QoS information fields with the current QoS conditions experienced by that node. The packet is dropped if QoS requirements cannot be met any more, avoiding flooding the network unnecessarily. Hop count is included to prevent the formation of loops in the forwarding process.

The session is closed by its initiator with a session destruction message (SES_DESTROY). Upon receiving it, all nodes clean their tables, whereas member nodes forwarding multicast data also free their resources allocated for that session. A node receiving a session destruction packet forwards it if it has also forwarded the corresponding initiation packet or is currently forwarding session data to at least one active session member.

C. Membership Management

A node wishing to join a session can directly do this if it is already a forwarding node in that session. Otherwise, it has to issue a join request (JOIN_REQ) containing the session information and the selected QoS level. The predecessors of the requester, which are aware of the session, propagate it upstream as long as QoS is satisfied. Ad hoc networks are highly dynamic, and available resources may change after the arrival of the QoS conditions with the session initiation packet. As explained in the following section, greeting messages are exchanged between neighbours to update nodes on the bandwidth usage in a neighbourhood. However, nodes do not send session status updates to avoid excessive control traffic. Instead, QoS is announced once by the session initiator and updated on demand. All nodes maintain request tables (TBL_REQUEST) to keep track of requests and replies they have forwarded to prevent false or duplicate packet processing.

The forwarded request eventually reaches some nodes which are already members of that session and can directly send replies (JOIN_REP). Members of a session are the initiator, forwarders, and receivers. Downstream nodes, having initiated or forwarded join requests, aggregate the replies they receive and forward only the one offering the best QoS conditions towards the requester, which also selects the reply with the best QoS conditions. It changes its status from predecessor to receiver (MCN_RCV) and sends a reserve message (JOIN_RES) to the selected node which has forwarded the best reply.

Upon receiving the reserve packet, intermediate nodes which are among the forwarding nodes of the requester change their status from predecessor to forwarder (MCN_FWD). They reserve resources, update their membership tables to keep a list of successors for that session, and forward the message upstream.

Eventually, the reserve message reaches the originator of the reply, which can be the session initiator with some or without any members, a forwarder with one or more successors, or a receiver. If the replier is the session initiator and this is its first member, it changes its status from initiator to server (MCN_SRV). If it is a receiver, it becomes a forwarder. In both cases, the replier records its successor in its member table and reserves resources to start sending multicast data. If the node is an active server or forwarder, it must have already reserved resources. It only adds the new member to its member table and continues sending the regular multicast data. At the end of each successful request-reply-reserve process, intermediate nodes have enough routing and membership data available to take part in the multicast data forwarding task.

A node needs to inform its forwarder on the multicast graph upon leaving a session. After receiving a quit notification (SES_LEAVE), the forwarding node deletes the leaving member from its member table. If this has been its only successor in that session, the forwarding node checks its own status. If the node itself is not a receiver, it frees resources and notifies its forwarder of its own leave.

D. Neighbourhood Maintenance

Each node periodically broadcasts greeting messages (NBR_HELLO), informing its neighbours on its existence and bandwidth usage, which is determined by the QoS classes and levels of the sessions being served or forwarded by that node. Greeting messages can be piggybacked to other control and data messages to reduce overhead. This way, nodes only need to send explicit greeting messages if they have not piggybacked any for a certain period of time. Each node aggregates the information it receives in its neighbourhood table (TBL_NEIGHBOUR). This table is used to calculate the total bandwidth currently allocated to multicast sessions in the neighbourhood, which is the sum of all used capacities of the neighbouring nodes for that particular timeframe. A node can only use the remaining capacity not used by itself and its immediate neighbours. Neighbourhood tables also help nodes with their decisions on packet forwarding. If a node does not receive any greeting messages for some time from one of its neighbours, it considers that neighbour lost.

IV. PERFORMANCE EVALUATION

The simulations are conducted using OPNET Modeler 10.0 Educational Version and Wireless Module [14]. The usage scenarios consist of open-air occasions such as search and rescue efforts and visits to nature in an area with boundaries, where a wired network infrastructure is not available. AQM nodes are modelled in three layers with application, session, and network managers. The application manager is responsible for selecting the type of application to run, setting its QoS requirements, and deciding whether to initiate, destroy, join or leave a session. The session manager is responsible for declaring new sessions initiated by its application manager to other nodes, sending requests for sessions its application manager wishes to join, keeping lists of sessions, members and other nodes' requests, processing and forwarding their information messages, and taking part in their join processes when necessary. The network manager is responsible for packet arrival and delivery, as well as for broadcasting periodic greeting messages to derive free bandwidth information.

TABLE I
QOS CLASSES AND REQUIREMENTS

QoS Class	Bandwidth Requirement	Delay Tolerance	Application Type
1	128 Kbps	10 ms	High-quality voice
2	256 Kbps	100 ms	CD-quality streaming audio
3	3 Mbps	90 ms	SDTV-quality streaming video
4	4 Mbps	10 ms	High-quality video conference

TABLE II
SIMULATION PARAMETERS

Parameter Description	Value
Area size	400 m x 400 m
Greeting message interval	10 s
Maximum available bandwidth	10 Mbps
Membership duration	600 s
Node distribution	Uniform
Node idle time	Exponential (600 s)
Session duration	1,200 s
Session generation / joining ratio	1 / 9
Simulation duration	8 h
Wireless transmission range	200 m

Multiple simulations are run in for each one-class network. Sessions belong to one of the four QoS classes defined in Table I. Simulation parameters are given in Table II.
Previous research efforts have essentially been evaluated through the use of several important ratios which give a notion about the internal efficiency of the protocol developed. These are data delivery ratio in terms of data bytes or packets sent, and control overhead in terms of control bytes or packets sent, all measured per data byte or packet delivered [15]. However, the main concern of this work is to evaluate AQM's efficiency in providing multicast users with QoS support and satisfying application requirements. Therefore, a new criterion is introduced in order to interpret the performance of the proposed scheme. Thus, the overload prevention grade O_{Member} is defined for multicast members to evaluate the node-level success ratio of AQM, and formulated as follows:

$$O_{Member} = 1 - \frac{o}{s + \alpha f} \quad (1)$$

In (1), o is the number of overloaded nodes, which have decided to serve or forward more sessions than is possible without exceeding the maximum available bandwidth. s is the total number of session servers, and f is the total number of forwarders. O_{Member} represents the ability of AQM to decrease the overload impact on the network, which gives the ratio of overloaded nodes to all serving and forwarding nodes. Since forwarders are intermediate nodes, which can be involved in more than one session concurrently, an overloaded forwarder has a greater impact on member satisfaction than an overloaded server. To reflect this difference, f is multiplied by a coefficient α, which is set to 4 in the simulations.
Figure 1 compares the overload prevention grades of AQM to the non-QoS scheme for class 2 applications. In AQM, nodes do not accept more traffic than they can handle. Overloaded members still occur due to the hidden terminal problem mentioned in Chapter III. However, their impact is limited. Overload prevention grades are above 95% in small networks of up to 30 nodes, and still above 85% for 50 nodes. The performance of AQM is significantly better than the non-QoS scheme. In the latter, nodes do not check their bandwidth before replying to join requests. Thus, more serving and forwarding nodes become overloaded than AQM, which affects all their successors in all the sessions they serve. As the network grows, more sessions are initiated, and more requests are accepted per node, which causes a drastic decrease in the member overload prevention ability of the non-QoS network.
Figure 2 shows that AQM outperforms the non-QoS scheme also for class 3 with higher bandwidth demands. It achieves 90% overload prevention in small networks, and remains around 75% for larger ones. Thus, it can be concluded that QoS support improves member efficiency significantly during multicast sessions.

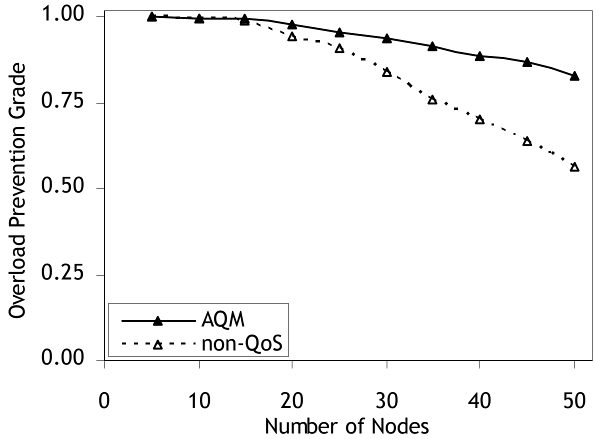

Fig. 1. Member overload prevention grades for QoS class 2.

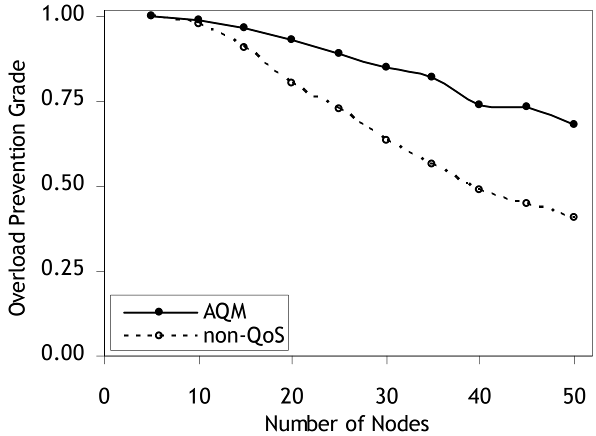

Fig. 2. Member overload prevention grades for QoS class 3.

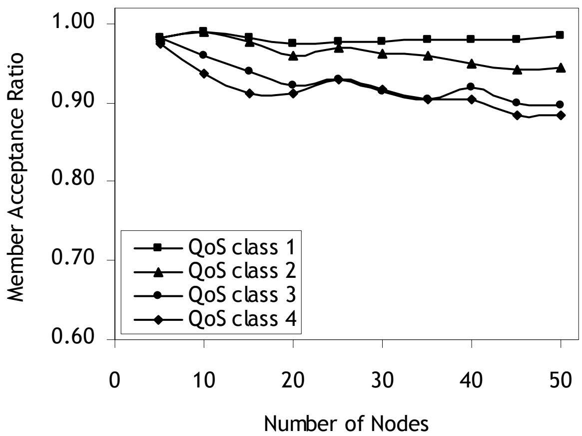

Fig. 3. AQM member acceptance ratios for all QoS classes.

The QoS-related multicasting decisions made by AQM cause some join requests to be rejected during the sessions. The ratio of successful joins to all requests, defined as the member acceptance ratio A_{Member}, is formulated as follows:

$$A_{Member} = \frac{r}{q} \quad (2)$$

In (2), r is the number of receivers, which are actually accepted to multicast sessions, and q is the total number of join requests issued by the nodes.

Figure 3 shows the member acceptance ratios of AQM. While improving the member overload prevention grades considerably, AQM also manages high member acceptance rates, which are above 90% for all QoS classes. Thus, it does not cause a significant decrease in accepted requests.

V. CONCLUSION

The growing user demand for group-oriented, high quality, mobile multimedia communication necessitates ad hoc QoS multicasting. AQM improves multicasting efficiency by checking resource availability within each node's neighbourhood based on previous reservations. AQM has a flat network structure where all nodes are equal. Thus, it avoids complicated topologies such as hierarchical or clustered network structures, which are hard to design and maintain. On the other hand, it is also possible to adapt AQM to a clustered network to scale with network size. Intra-cluster multicasting can be handled by AQM, whereas inter-cluster communication is managed by its higher-layer, hierarchical version. Mobile networks cannot afford a pure on-demand scheme for QoS support. AQM proposes a hybrid multicasting method with table-driven session management and on-demand QoS verification upon the initialisation of a join process.

The primary evaluation criteria for AQM are overload prevention and member acceptance. Simulation results show that, by applying QoS restrictions to the ad hoc network, AQM significantly improves member overload prevention grades, while keeping acceptance ratios high. While the application of QoS rules causes some users to be rejected, the lack of these restrictions causes much more users to be affected by overloaded nodes, especially for larger networks and higher bandwidth requirements. Thus, AQM proves that QoS is essential for and applicable to ad hoc multimedia networks.

An important research topic for AQM is keeping the QoS data up-to-date, and involves handling of lost neighbours, on-demand data exchange, and interpretation of changes in a node's QoS status. A point closely related to QoS data accuracy is the hidden terminal problem. To overcome it, an extension to the request-reply-reserve process is necessary, whereby each replying node consults its neighbourhood to see if there are any objections. Another issue is the adaptation of multicast sessions to changing QoS conditions. The MAC layer is responsible for resource reservation and the acquisition of available link bandwidth information, which is another significant task since it involves infrastructure decisions. Within the scope of this work, however, AQM's integrity has been maintained by addressing these issues in higher-layer algorithms.

ACKNOWLEDGEMENTS

This work was supported in part by the State Planning Organisation, Turkey, under Grant No. DPT98K120890 and OPNET Technologies University Research Program.

REFERENCES

[1] Van Nee, R, and R. Prasad, *OFDM for Wireless Multimedia Communications*, Artech House Publishers, 2000.

[2] Walrand, J, and P. Varaiya, *High-Performance Communication Networks*, 2nd edition, Morgan Kaufmann Publishers, 2000.

[3] Chakrabarti, S, and A. Mishra, "QoS Issues in Ad Hoc Wireless Networks," *IEEE Communications Magazine*, Vol. 39, No. 2, pp. 142-48, February 2001.

[4] Lee, S.J, M. Gerla, and C.C. Chiang, "On-Demand Multicast Routing Protocol," *Proceedings of IEEE WCNC 1999*.

[5] Lee, S.J, W. Su, J. Hsu, and M. Gerla, "On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks," *IETF MANET Working Group Internet Draft* (draft-ietf-manet-odmrp-02.txt), work in progress, January 2000.

[6] Royer, E.M, and C.E. Perkins, "Multicast Operation of the Ad-Hoc On-Demand Distance Vector Routing Protocol," *Proceedings of ACM/IEEE MOBICOM 1999*.

[7] Royer, E.M, and C.E. Perkins, "Multicast Ad Hoc On-Demand Distance Vector (MAODV) Routing," *IETF MANET Working Group Internet Draft* (draft-ietf-manet-maodv-00.txt), work in progress, July 2000.

[8] Ozaki, T, J.B Kim, and T. Suda, "Bandwidth-Efficient Multicast Routing for Multihop, Ad Hoc Wireless Networks," *Proceedings of IEEE INFOCOM 2001*.

[9] Ji, L, and M.S. Corson, "Differential Destination Multicast – A MANET Multicast Routing Protocol for Small Groups," *Proceedings of IEEE INFOCOM 2001*.

[10] Ji, L, and M.S. Corson, "Explicit Multicasting for Mobile Ad Hoc Networks," *Mobile Networks and Applications (MONET)*, Vol. 8, No. 5, pp. 535-549, October 2003.

[11] Lee, S, and C. Kim, "A new wireless ad hoc multicast routing protocol," *Elsevier Science Computer Networks*, Vol. 38, pp. 121-135, February 2002.

[12] Toh, C.K, *Ad Hoc Mobile Wireless Networks: Protocols and Systems*, Prentice Hall PTR, 2002.

[13] Li, W, "Overview of fine granular scalability in MPEG-4 video standard, " *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 11, No. 3, pp. 385-398, March 2001.

[14] OPNET Technologies Inc, Bethesda, MD, USA, Available at http://www.opnet.com

[15] Corson, S, and J. Macker, "Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations," *RFC 2501*, January 1999.