Performance evaluation of a mesh-evolving quality-of-service-aware multicast routing protocol for mobile ad hoc networks

Bür, Kaan; Ersoy, Cem

Published in:
Performance Evaluation

DOI:
10.1016/j.peva.2009.08.008

2009

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

Performance Evaluation of a Mesh-Evolving Quality-of-Service-Aware Multicast Routing Protocol for Mobile Ad Hoc Networks

Kaan Bür[1], Cem Ersoy

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

Abstract

The tremendous amount of multimedia applications running across the wireless communication medium makes quality of service (QoS) a fundamental requirement for mobile ad hoc networks. However, it is not easy to incorporate QoS into these networks. Moreover, the growing number of group-oriented applications also necessitates the efficient utilisation of network resources. The multicast model is a promising technique which can achieve this efficiency by facilitating the inherent broadcast capability of the wireless medium. The mesh-evolving ad hoc QoS multicast (MAQM) routing protocol is developed to address the resource efficiency and QoS problems with one, integrated solution. MAQM achieves multicast efficiency by tracking the availability of resources for each node within its neighbourhood. The QoS status is monitored continuously and announced periodically to the extent of QoS provision. Using these features, MAQM nodes can make their decisions on joining a new multicast session based on the sustainability of their perceived QoS. MAQM also evolves the initial multicast tree into a mesh during the course of an ongoing session to achieve a more robust network topology. Thus, MAQM integrates the concept of QoS-awareness into multicast routing in mobile ad hoc networks. Since ad hoc networks require the protocol control overhead to be as small as possible, we analyse the multicast session establishment process of MAQM to see its impact on the protocol performance in terms of system control overhead. We also evaluate the performance of MAQM through computer simulations using various qualitative and quantitative criteria. The simulation results validate our mathematical analysis of the control overhead and show that MAQM significantly improves multicast efficiency through its QoS-aware admission and routing decisions with an acceptably small overhead. Thus, MAQM shows that QoS is not only essential for, but also applicable to mobile ad hoc networks.

Keywords: Mobile ad hoc networks, multicast routing, quality of service, control overhead analysis, performance evaluation, wireless communications.

1. Introduction

Mobile ad hoc networks possess unique properties, which make them very different from traditional wired and infrastructure-based wireless systems. It is a significant technical challenge to provide reliable high-speed end-to-end communications in mobile ad hoc networks, due to their dynamic topology, distributed management and multihop connections. Moreover, the actual throughput of wireless communications is often much less than the maximum radio transmission rate, due to the effects of multiple access, fading, noise and interference conditions [1]. These effects result in time-varying channel capacity, making it difficult to determine the aggregate bandwidth between two endpoints. Finally, in addition to bandwidth, other network resources like energy, processing power and memory, which are relatively abundant in wired environments, are also strictly limited and have to be preserved in mobile ad hoc networks.

For these reasons, it is very important for ad hoc networks to design efficient methods of conserving the scarce resources, while not incurring too much control overhead. Multicasting is a promising technique to provide a subset of network nodes with the service they demand while not jeopardizing the resource requirements of others. As a result of their broadcasting capability, the nodes of an ad hoc network are inherently ready for multicasting.

In order to meet the qualitative expectations of its users, on the other hand, ad hoc networks also need support for multimedia, which makes quality of service (QoS) a fundamental requirement. Unfortunately, it is not an easy task to incorporate QoS to ad hoc multicast routing. Protocols designed for conventional wired networks are not appropriate for ad hoc networks due to their lack of adaptation to the unpredictable network topology and excessive overhead [2]. Wireline QoS algorithms rely on the availability of precise state information. However, this information is inherently imprecise in ad hoc networks, where nodes join, leave and rejoin the network at any place, and links appear or disappear any time.

Manuscript submitted December 1, 2008, to the special issue on *"Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks"* of Elsevier's *Performance Evaluation*. Revised version submitted June 30, 2009.

This work was supported in part by State Planning Organisation, Turkey, under the grant numbers DPT98K120890 – DPT03K120250, and by the university research program of OPNET Technologies Inc, USA.

[1] Corresponding author. Now affiliated to Broadband Communications Group, Department of Electrical and Information Technology, Lund University, Sweden. Phone: +46-46-2227341. E-Mail: kaan.bur@eit.lth.se

The ad hoc QoS multicast (AQM) routing protocol [3] integrates the concept of QoS-awareness into multicast routing in mobile ad hoc networks as a composite solution to the problems stated above. AQM tracks QoS availability within each node's neighbourhood based on current bandwidth reservations made for ongoing sessions and the requirements reported by the neighbours. It achieves delay restriction by limiting the number of hops allowed to join a session with specific QoS requirements. Using these resource and hop distance tracking features, AQM nodes can make their decisions on sending join requests or replies based on the availability of QoS. Nodes are prevented from applying for membership if there is no QoS-satisfying path to reach the multicast session tree.

Two types of network topologies, presenting different performance characteristics, are generally identified for ad hoc multicast routing protocols. While tree-based protocols use resources more efficiently and incur less overhead, mesh-based protocols are more robust to the topological changes in the network [4]. Thus, in this article, we enhance AQM with new algorithms that evolve multicast trees into meshes to improve its data delivery rate and robustness. We call the new version the mesh-evolving ad hoc multicast (MAQM) protocol, and evaluate its performance through a mathematical analysis and extensive computer simulations. As mentioned above, it is desirable for mobile ad hoc networks that sessions are managed with an overhead as small as possible. Our analysis investigates MAQM's session joining process, and shows that this overhead is acceptable. Our performance evaluation validates through computer simulations that MAQM improves multicast routing efficiency for members and sessions significantly through QoS management with an acceptable control overhead, which degrades gracefully throughout a session.

The rest of this article is organised as follows: Section 2 categorises previous research and summarises some recent advances in ad hoc QoS multicast. Section 3 introduces MAQM with special emphasis on tree-to-mesh evolution. An analysis of the control overhead, considering the multi-hop, omnidirectional nature and limited transmission range of mobile ad hoc communication, is provided in Section 4. Novel, qualitative and measurable performance metrics are defined and simulation results are presented in Section 5 to evaluate the performance of MAQM, whereby a mobile ad hoc network supporting multiple QoS classes is simulated for a realistic usage scenario. Finally, Section 6 concludes the article and provides some future research directions.

2. Background of Ad Hoc Multicast Routing

There are many applications of mobile ad hoc networks that involve point-to-multipoint or multipoint-to-multipoint communication patterns, such as multimedia streaming, video conferencing, database management, distributed computation and real-time workgroup activities. This makes multicast routing a necessity to provide connection and coordination among a given set of nodes. It is particularly advantageous to facilitate multicast rather than use multiple unicast in ad hoc networks, where scarce resources are shared in the wireless medium and it is particularly important to reduce the transmission overhead. However, conventional wired network multicast protocols do not perform well in the ad hoc domain due to the unreliable nature of the wireless links and the dynamic network topology. These protocols usually require global knowledge on routing such as link state or distance vector structures, which are not feasible for ad hoc networks [2]. Therefore, many ad hoc multicast routing protocols are proposed to address the problem.

2.1. A Classification of Ad Hoc Multicast Protocols

Multicast routing protocols for mobile ad hoc networks are classified into categories according to connectivity management (source- vs. receiver-initiated), operation and maintenance (proactive vs. reactive), and, most popularly, multicast topology (tree- vs. mesh-based) [2, 5]. In a tree-based multicast routing protocol, a node accepts packets only when they come from another node which a tree branch has been established with. Thus, there is only a single path between a sender-receiver pair. Tree-based protocols are further categorised into shared-tree [6-13] and source-tree [3, 14-22] topologies. In the former, all members of a multicast group, including the sources, are connected via a single shared tree, whereas in the latter, a group consists of members scattered on multiple trees rooted at their respective sources. A shared multicast tree spanning all the group members may not be optimal for the individual sources, but they are more scalable when the number of sources in a session or the number of multicast sessions increases. One common technique used with this approach is to assign a node, known as the rendezvous point or the core, to accept join requests from members. The multicast connection then consists of shortest paths from the core to each of the members. On the other hand, source-tree-based protocols perform better than shared-tree-based protocols under heavy traffic since they achieve more efficient load balancing. Since there is only a single path between senders and receivers in tree-based multicast routing protocols, they are vulnerable to the dynamics of the ad hoc network such as node mobility and link breaks. In contrast, mesh-based multicast protocols [23-33] maintain a mesh consisting of a connected component of the network containing all the receivers of a group. They construct a mesh that allows data packets to be transmitted over more than one path from a sender to a receiver to increase robustness at the price of transmission redundancy.

2.2. QoS in Ad Hoc Multicast Routing

Numerous protocols are developed to build and maintain a multicast graph and perform routing in mobile ad hoc networks. A selected chronology is given in Table 1. However, these protocols do not address the multimedia QoS requirements of sessions within the framework of ad hoc multicast routing, which is becoming increasingly important as the demand for mobile multimedia increases. Thus, a selection of more recent efforts, which attempt to deal with the QoS problem, is presented in this section.

Table 1. Chronology of ad hoc multicast routing protocols.

Year	Protocol Name	Topology	Initiator	Flooded Control	Periodic Control
1998	AMRoute [6]	Shared tree	Source	Yes	Yes
	ASTM [7]	Shared tree	Receiver	Yes	Yes
	FGMP [23]	Mesh	Receiver	Yes	Yes
	AMRIS [8]	Shared tree	Source	Yes	Yes
1999	CAMP [24]	Mesh	Receiver	No	No
	MAODV [9]	Shared tree	Receiver	Yes	Yes
	MCEDAR [14]	Source tree	Receiver	Yes	No
	ODMRP [25]	Mesh	Source	Yes	Yes
	BEMR [15]	Source tree	Receiver	Yes	No
2000	DDM [16]	Source tree	Receiver	Yes	Yes
	NSMP [26]	Mesh	Source	Yes	Yes
	ABAM [17]	Source tree	Source	Yes	No
2001	MZR [18]	Source tree	Source	Yes	Yes
	MMA [10]	Shared tree	Receiver	No	Yes
2002	SRMP [27]	Mesh	Receiver	Yes	No
	DCMP [28]	Mesh	Source	Yes	Yes
	WBM [19]	Source tree	Receiver	Yes	No
2003	RDG [29]	Mesh	Receiver	Yes	Yes
	PLBM [20]	Source tree	Receiver	No	Yes
2004	AQM [3]	Source tree	Receiver	No	Yes
	ODQMM [11]	Shared tree	Receiver	Yes	Yes
	PUMA [30]	Mesh	Receiver	No	Yes
2005	MANSI [12]	Shared tree	Receiver	Yes	Yes
	QMRPCAH [13]	Shared tree	Receiver	No	Yes
	HVDB [31]	Mesh	Source	No	Yes
	OPHMR [32]	Mesh	Source	Yes	Yes
2006	MLCT [21]	Source tree	Source	Yes	No
	QAMNet [33]	Mesh	Source	Yes	Yes
2007	ABMRS [22]	Source tree	Source	Yes	Yes

On-demand QoS multicast for MANETs (ODQMM) integrates bandwidth reservation into routing and tries to identify bandwidth availability while searching for the path [11]. It is based on the multicast ad hoc on-demand distance vector (MAODV) routing protocol [9]. It inserts flags into the control packets to indicate requests for resource reservation. Two reservation styles, fixed and shared, are defined, which correspond to video and audio communications, respectively. Any node wishing to join a multicast group has to reserve the necessary amount of bandwidth. However, ODQMM limits all reservations to the amount requested by the server and assumes that the bandwidth information is provided by the underlying layer.

QoS multicast routing protocol for clustering mobile ad hoc networks (QMRPCAH) utilises a hierarchical topology provided by clusters and bridges [13]. Local nodes within a cluster send their join requests with QoS metrics to their intercluster bridge nodes that know the other bridges containing nodes on the same multicast tree. If the tree is unknown to the bridge, the request propagates to the next upper bridge in the hierarchy. If the tree does not exist, the local node is informed and has to generate it. Resources are reserved by the intermediate nodes during the request and are freed if the join process times out.

The multiple least-cost trees (MLCT) approach finds multiple paths or trees for a multicast connection to meet its bandwidth requirement by aggregating the resources on these paths [21]. Through a route request/reply process, information on all possible routes to all destinations within the delay range is gathered. A path/tree selection algorithm then finds multiple paths/trees to satisfy QoS in terms of bandwidth and delay. Three alternate selection criteria are introduced to choose multiple paths based on shortest-path trees, least-cost trees or even multiple least-cost trees.

Hypercube-based virtual dynamic backbone (HVDB) is a model that defines new QoS requirements for mobile ad hoc networks, such as high availability and good load balancing [31]. The hypercube architecture provides fault tolerance and a small diameter to address the first issue, and regularity and symmetry for the second. Nodes are grouped into clusters, clusters into hypercubes and hypercubes into a mesh to build logical routes. Each node sends its local membership information to the cluster head, which shares it with other cluster heads in a hypercube. The multicast tree is built at the mesh tier by using the shared information and unicast to the cluster heads in other hypercubes with group members. Each cluster head sends the information to other cluster heads in the same hypercube via local logical routes. Data packets are sent to group members by local broadcast.

QoS-aware mesh network (QAMNet) is an extension to mesh-based protocols with resource-aware admission control interweaved with the mesh creation process [33]. Senders flood the network with new session information. Intermediate nodes update a bottleneck bandwidth field in the messages to report on the resource status. QoS flows are admitted if enough resources are available during mesh creation. Best-effort flows are regulated by a traffic shaper.

3. Mesh-Evolving Ad Hoc QoS Multicast

Although resource management schemes are developed to serve ad hoc routing and multicast protocols as QoS modules [34], there has not been an ad hoc multicast protocol that incorporates QoS directly in its admission and routing decisions. AQM is presented as a solution to this problem [3]. In this section, we introduce MAQM, our new version evolving the initial multicast trees into meshes to improve connectivity and robustness. We first explain its session and membership management, mobility adaptation, and resource allocation procedures. Then, emphasis is laid on the logic behind and procedures related to the evolution of the initial multicast tree into a mesh.

3.1. Session Initiation

A new session is announced by its server that broadcasts an initiation packet with the session's QoS information, which contains the bandwidth and hop count requirements to join it. A table of active sessions is maintained at each node to keep the information on session definitions. Using these tables, nodes spot new sessions and forward their initiation packets. A membership table is used by each node to denote whether QoS is supported on the path from the session server up to this node. The hop count information in the packet is used to prevent loops in the forwarding process. The session initiation packet is forwarded as long as the QoS requirements in terms of bandwidth and delay are met. The packet is dropped if QoS requirements cannot be satisfied any more to avoid flooding. The session information is refreshed periodically via session update packets sent by the session server. Similar to the session initiation packets, they are propagated throughout the network as long as the QoS requirements of the session can be fulfilled. Figure 1 [3] shows an example for the session initiation process.

3.2. Session Membership

When a node broadcasts a join request for a session, its predecessors propagate the packet upstream as long as QoS, in terms of bandwidth and delay, can be satisfied. They maintain a request table to keep track of the requests and replies they have forwarded and prevent false or duplicate packet processing. To keep the size of the overhead under control, they only deal with the requests for the sessions made known to them through announcements. A forwarded request eventually reaches members of that session which issue replies back to the requester if QoS can be satisfied. Downstream nodes, which have forwarded the join requests, now forward the replies towards the requester. Among the replies it receives, the originator of the join request selects the one with the best QoS conditions. It changes its status from predecessor to receiver and sends a reserve message to the selected node, which propagates along the selected path and finally reaches the originator of the reply. Intermediate nodes on the path become forwarders; they delete the request and the replies to it from their tables. Figure 2 [3] shows an example for the three-phase session join process.

In MAQM, after being accepted to a session, the receivers constantly monitor their own perceived QoS level, such as the average data arrival and packet loss rates. If a member experiences loss or delay beyond the limits of acceptable QoS, it decides that the minimum QoS required for the session cannot be sustained anymore. Thus, after notifying its forwarder, the member drops itself off the session to prevent further waste of resources. It should be noted that MAQM adopts the QoS approach known as *'soft QoS'*, i.e., QoS is supported in a statistical sense, without a 100% guarantee for all session members all the time.

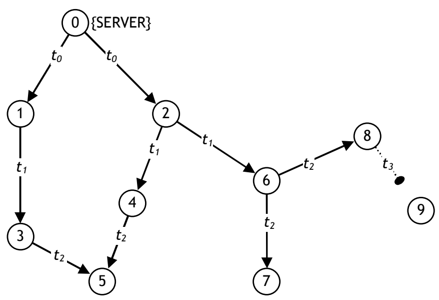


Figure 1. The MAQM session initiation process: An initiation message is broadcast by n_0, the server. It propagates through the network as long as its QoS requirements are met, informing all nodes from n_1 to n_8, which update their session and membership tables. n_9 is not informed since it is beyond the QoS limits in terms of hop count. $t_i < t_{i+1}$, $0 \le i \le 3$, represent the relative timing of the messages.

3.3. Mobility Adaptation

One of the major concerns for ad hoc networks is the ability of the routing infrastructure to cope with node mobility. In order to maintain connectivity within their neighbourhood as well as to support QoS with maximum accuracy and minimum overhead under node mobility, MAQM performs periodic maintenance operations.

Each node periodically broadcasts greeting messages, informing its neighbours on its existence and bandwidth usage, which is determined by the QoS requirements of the sessions being served or forwarded by that node. Greeting messages can be piggybacked to other control and data messages to reduce control overhead. Each node aggregates the information it receives with these messages in its neighbourhood table. 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 time frame.

If a node does not receive any greeting messages from a neighbour for a while, it considers that neighbour lost. Lost neighbours are marked as such for a predefined short period of time, at the end of which they are deleted from the neighbourhood table if they do not reappear. To prevent unnecessary message exchanges, nodes need to detect new neighbours quickly and distinguish them from the neighbours reappearing after a short period of time and not necessitating any update. If the lost neighbour is related to a session, it is also removed from the session, membership and request tables. This is an essential operation to keep the nodes up-to-date regarding the sessions and ready for future membership management activities such as initiating a new join request or replying to other nodes' join requests.

Additional action can be necessary depending on the status of the lost neighbour as well as that of the node itself. When an active session member, e.g. a receiver, loses its only upstream forwarder, this means that it loses its connection to the session. In this case, it restarts the session

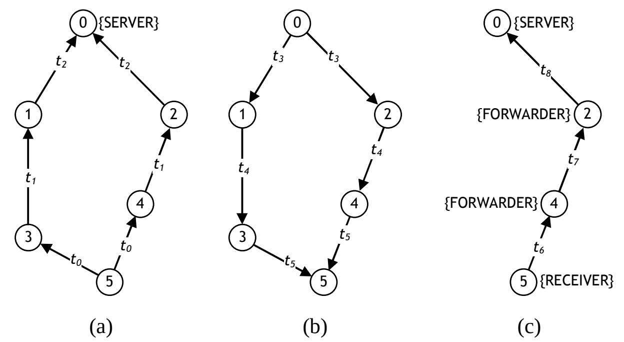


Figure 2. The MAQM session joining process: (a) A join request is issued by n_5. It propagates towards any member of the session as long as QoS can be satisfied. Nodes from n_1 to n_4 update their request tables and forward the packet since they are not session members. (b) A reply is sent back from n_0 to n_5. It is forwarded by n_1, n_2, n_3, n_4. (c) n_5 sends a reservation along the selected QoS path via n_4, n_2, n_0, which reserve resources and update their status. $t_i < t_{i+1}$, $0 \le i \le 8$, represent the relative timing of the messages.

join process. It also informs its successors with a lost session message if it is a forwarding member of the session. By doing this, it lets them know that it should be deleted from the list of forwarders for that session. Downstream nodes receiving the lost session messages interpret them in a similar way to update their status regarding the lost session and forward the message if necessary. This mechanism, combined with the periodic updates mentioned previously, keeps nodes up-to-date regarding the QoS status of the sessions and prevents them from making join requests that are infeasible in terms of resource allocation.

3.4. Resource Management

The streaming nature of multimedia applications necessitates a pipelined approach to checking resource availability. Concerning a session server about to allocate resources for its first member, twice as much bandwidth has to be available in the neighbourhood than the amount required by the QoS requirements of the session. The forwarding node immediately following the server on the path to the member belongs to the same neighbourhood as the server and shares the same free bandwidth. Therefore, a session server has to ensure that its successor also has enough bandwidth available to forward multicast data packets that it receives. On the other hand, a forwarder has to deal with its predecessor as well as its successor. Once the multicast session starts, it receives packets from its predecessor, rebroadcasts them, and allows its successor to forward the packets further downstream. Therefore, an intermediate node about to take part in the packet forwarding process has to check for availability of three times as much bandwidth than the amount needed by the session, since it shares the available bandwidth of the same neighbourhood as its immediate predecessor as well as its immediate successor. Thus, nodes have to check for availability of the necessary bandwidth according to their position within the multicast tree before accepting a request. When it is time to allocate resources, on the other hand, each node allocates only the amount of bandwidth that is required for its individual transmission. Figure 3 shows an example of this approach to resource management, which we call a virtual tunnel of bandwidth.

3.5. Tree-to-Mesh Evolution

One of the major concerns with mobile ad hoc networks is that they have to operate in an environment of continuous topological changes. The mobility and limited transmission range of the nodes cause their wireless links to break very frequently, leading to disconnections and data loss. Thus, it is particularly important that MAQM improves its robustness against topological changes. This can be done if extra wireless links can be added to the initial multicast tree, evolving it into a mesh during the course of data streaming in an intelligent way, such that member connectivity is strengthened without compromising efficiency in resource usage. MAQM utilises the inherent broadcast capability of the ad hoc network to achieve this goal.

As a result of broadcasting, receivers can start receiving multicast data from nodes other than their forwarders. This means that there is actually an active session member within the transmission range other than the original data forwarder selected for the session. In this case, the receiver decides that it can use this node, streaming multicast data actually to a third node, as an extra forwarder in order to improve the chance of remaining connected to the session despite frequent topological changes. The receiver informs the additional forwarder of its existence by sending one single-hop registration message to this node for that session and, thus, ensures that the forwarder is also aware of its new receiver. Figure 4 shows an example for such an evolution.

The operation described above increases the probability of connectivity and decreases the frequency of reconnection attempts, not only for the node itself but also for any other receivers further downstream, if the node doing the

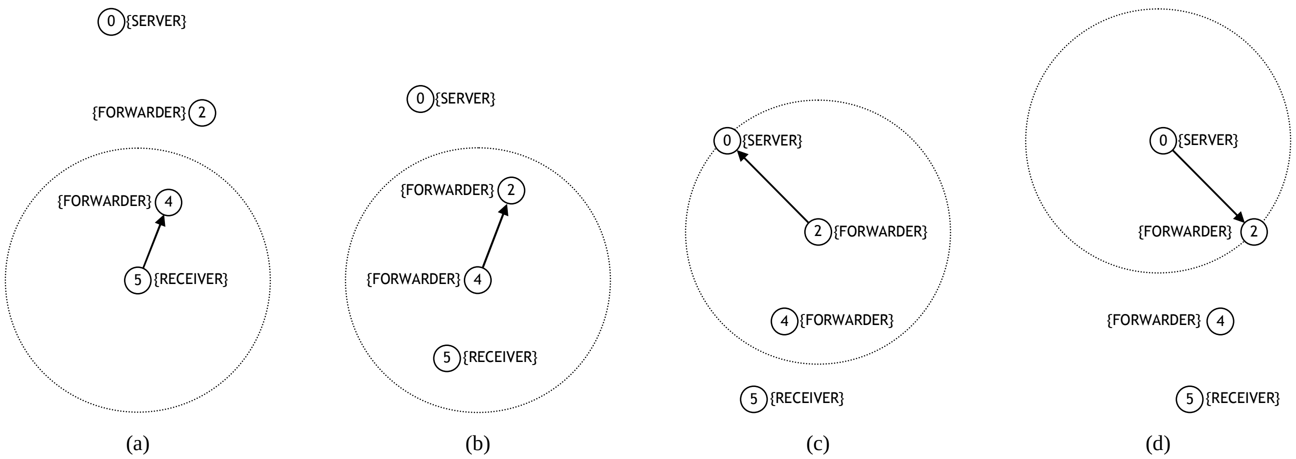


Figure 3. The virtual tunnel approach to bandwidth availability: (a) n_5 wants to join the session of n_0. It checks for the QoS bandwidth to make sure that, later, it can receive data. It sends a request. (b) Upon receiving the request, n_4 checks for two times QoS bandwidth since it has to receive and forward data packets in case n_5 actually joins the session. (c) The request propagates further upstream. n_2 checks for three times QoS bandwidth since, in addition to its predecessor n_0 and itself, it has to make sure that n_4 can also forward the streaming data downstream. (d) Finally, n_0 checks for two times QoS bandwidth since, as the server, it sends data packets and then lets n_2 forward them.

operation is at the same time a forwarder. In other words, through the addition of extra links with existing forwarders, which leads to fewer session losses and increases members' robustness, the general session satisfaction is improved.

The addition of extra links during data streaming has several benefits. First of all, forwarders and receivers are connected to their session servers through multiple intermediate forwarders, which makes session losses a less frequent event and yields to less control overhead. In other words, by increasing redundancy in their graphs, multicast sessions become less prone to lost forwarders. Secondly, a receiver registers itself at an additional forwarder only if this forwarder delivers fresh data packets, i.e., the additional forwarder yields an extra path shorter than the one initially selected by the receiver. These improvements lead to increased data delivery rates and decreased end-to-end delay, which means that more session members are satisfied by the provided QoS. Moreover, registration at more than one forwarder does not generate extra data traffic since these are already forwarding data to their existing receivers.

4. Performance Analysis of the Join Process

There are three groups of control packets in MAQM's protocol structure. The session module sends session initiation and termination packets once per session, which are relatively rare events. In addition, it also sends periodic session update packets. The overhead caused by these messages depends mainly on their frequency. Notifications for lost sessions are sent by this module as well, which is expected to happen more frequently due to node mobility. However, assuming that there are much less sessions than members in the network at any instant, it can be argued that the control packets generated by the session module are only a fraction of the ones generated by the membership module. The greeting messages of the network module are another group of periodic messages.

Various packets are sent by the membership module during the session joining process. The propagation of the request, reply and reserve messages depends on the hop distance between the originator of the request and the nearest session member on the multicast mesh, which can change with each request. It is therefore harder to perform an analysis of the control overhead incurred by the join requests. Since the join processes are expected to be the majority of all control events in the ad hoc network, however, they deserve this investigation.

In this section, the join process of MAQM and its impact on the system control overhead is analysed to provide an estimate of the amount of control messaging it incurs in the network. First, the attempt of the first member to join a session is examined. Upon the successful joining of the primary receiver, possible behaviour of subsequent join attempts is analysed. Finally, the three-phase join process is revisited for a single join attempt in isolation to observe the way a request propagates from its originator towards the session members and how it forces intermediate nodes to react to it. In the analysis, trees are used as the multicast session infrastructure. Since the evolution into meshes does not add new nodes to this infrastructure, analysis results with meshes would yield the same results. This point is explained further at the end of this section.

There are research efforts analysing the expected number of hops for a source to reach a destination [35]. The expected one-hop progress of a packet in the desired direction is defined as the distance between the sender and a receiver projected on a line connecting the source and the destination. It is formulated as a function of the number of neighbours, the node density, the transmission range and the distance from the source to the destination. The average number of hops between two nodes randomly placed within a circular area is then found by dividing their expected distance by the expected one-hop progress. However, the calculation of the average number of hops is not trivial and requires information on the Euclidian distance between nodes. Moreover, the analysis is aimed at unicast. Thus, a new approach is necessary, which explicitly considers the properties of multicast communication.

4.1. Primary Receiver of a Session

In the beginning, the multicast session consists of the server only, which is inactive since it has no receivers yet. Then, with the addition of the first receiver, a multicast tree

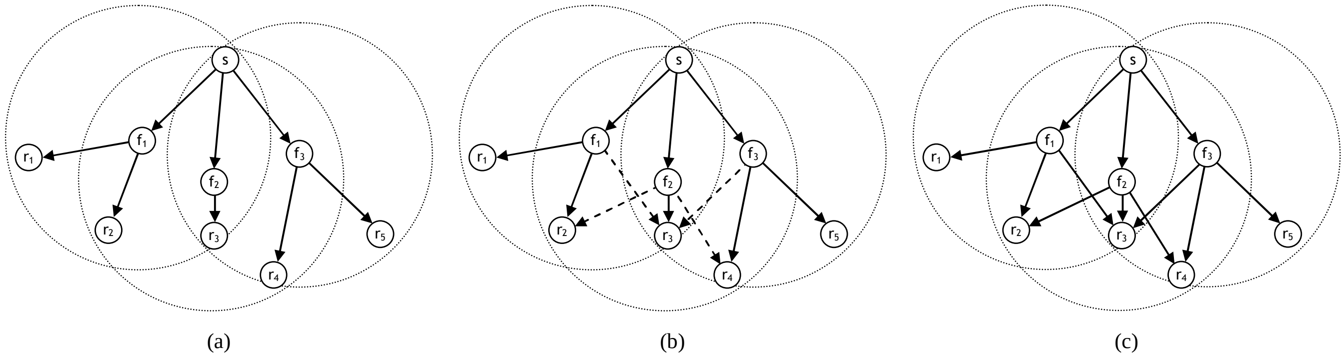


Figure 4. The evolution of the initial multicast tree into a mesh: (a) A session tree is initiated. (b) Receivers r_2, r_3, r_4 receive data packets from nodes other than their current parents and discover extra forwarders f_1, f_2, f_3. They register themselves with these. (c) Extra forwarders receive registration messages and enter their new receivers into their member tables. New links are established. The tree evolves into a mesh.

is initialised, which evolves into a mesh following the principles explained in Section 3.5. The finding of the session becomes increasingly easier for subsequent join candidates. Therefore, it is important to observe the events during the join process of the first candidate to the session.

Definition 1: Let the network area be of circular shape with the radius R. Let R be an integer multiple of r, the node transmission range. Let s be *the server of the session* located at the centre of the network area. Let m_1 be *the first receiver to join* the session.

Figure 5(a) depicts the area that corresponds to the one-hop neighbourhood of s. Under the assumption of uniform node distribution in two-dimensional space, the probability that m_1 is within the boundaries of this area is formulated as the ratio of the number of nodes in this area to the number of all the nodes in the network. The formula also gives the probability that m_1 is one hop away from s.

Definition 2: Let H_1 be a discrete random variable having a probability mass function $p(h) = P\{H_1 = h\}$, which is defined as the probability that m_1 reaches s in h hops. The probability that m_1 reaches s in one hop is:

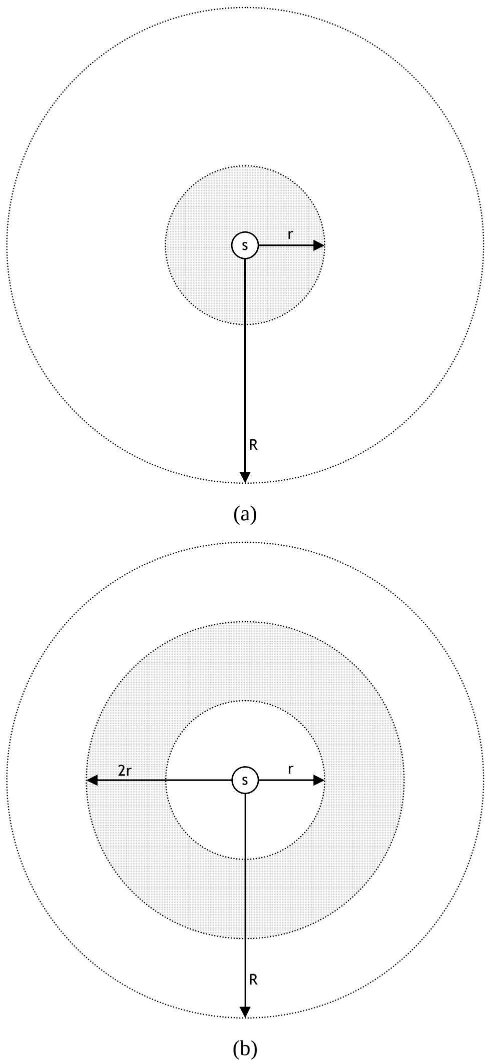


Figure 5. The one- and two-hop neighbourhoods of the server.

$$P\{H_1 = 1\} = \frac{r^2}{R^2} \tag{1}$$

Equation 1 shows that the desired probability, which is the ratio of the number of nodes within the respective areas, is equal to the ratio of the areas of the respective circles, which is equal to the ratio of the respective radii squared.

Figure 5(b) depicts the two-hop neighbourhood of s. Similar to the one-hop solution; the probability that the first candidate m_1 is exactly two hops away from s is equal to the ratio of the two-hop neighbourhood area of s to the area of the whole network.

$$P\{H_1 = 2\} = \frac{\pi\left[(2r)^2 - r^2\right]}{\pi R^2} \tag{2}$$

With the help of these results, the general case probability of m_1 being exactly in the h^{th} neighbourhood of s can be formulated as follows:

$$P\{H_1 = h\} = (2h-1)\frac{r^2}{R^2} \tag{3}$$

It should be noted that $h \leq \gamma$, which is the maximum number of hops that m_1 needs to take in order to reach s and is defined as follows:

$$\gamma = \frac{R}{r} \tag{4}$$

Definition 3: If H_1 is a discrete random variable having a probability mass function $p(h) = P\{H_1 = h\}$, then the expected value of the number of hops m_1 needs to reach s, which is the weighted average of $p(h)$, the possible values of H_1, is found, using Equation 3 and Equation 4, as follows:

$$E[H_1] = \sum_{h=1}^{\gamma} h(2h-1)\frac{r^2}{R^2}$$

$$E[H_1] = \frac{r^2}{R^2}\left(2\sum_{h=1}^{\gamma} h^2 - \sum_{h=1}^{\gamma} h\right)$$

$$E[H_1] = \frac{2}{3}\gamma + \frac{1}{2} - \frac{1}{6}\gamma^{-1} \tag{5}$$

Equation 5 shows that the expected number of hops for the first receiver m_1 to reach the multicast server s is mainly influenced by the ratio of the network radius R to the transmission range r. The result is important since it affects the overhead incurred during the join process of a receiver, which is analysed in Section 4.3. It also has an impact on the overhead caused by subsequent join requests made for the same session, which is analysed in Section 4.2.

As mentioned previously, the expected number of hops grows linearly with the ratio between the physical size of the network area and the transmission range. Thus, an ordinary multicast routing protocol without any QoS constraints experiences increasing overhead as this ratio increases, regardless of the density of the nodes, which is not a desirable property for the sake of scalability. On the other hand, the nodes that are far away from the server also suffer from high delays and packet losses, in addition to frequent disconnections, due to the length of the path to the server. Therefore, it is preferable that meshes with high diameter values are avoided. This is why MAQM applies hop count limitations as part of its QoS management strategies. The QoS requirements followed by MAQM separately for each session restrict γ in Equation 5, bounding the receivers by a virtual network border.

4.2. Subsequent Receivers

By joining the session, the first receiver establishes a connection with the server, which is the first path on the multicast tree before it evolves into a mesh with the number of intermediate nodes 1 less than the hop distance selected as the first set of forwarders. Based on the assumption of uniformly distributed nodes, the resulting initial multicast tree and its aggregated coverage area can be determined, which is the superposition of the transmission ranges of the current session members.

Definition 4: Let h_1 be *the hop distance* between the first receiver m_1 and the server s, such that $1 \le h_1 \le \gamma$.

An example is given in Figure 6(a) where h_1 equals 4 and the distance between the nodes is r. For the sake of simplicity, it is assumed that the nodes are located on a line. Thus, r being the upper limit of the distance between two consecutive nodes, the coverage area to be examined is the maximum possible. Similar to the probability calculations made for the first receiver, the probability that the second receiver joins the multicast mesh in one hop can be defined as the ratio of this area to the total area of the network. Figure 6(b) shows the intersecting areas to be calculated with their x_h values along the axis.

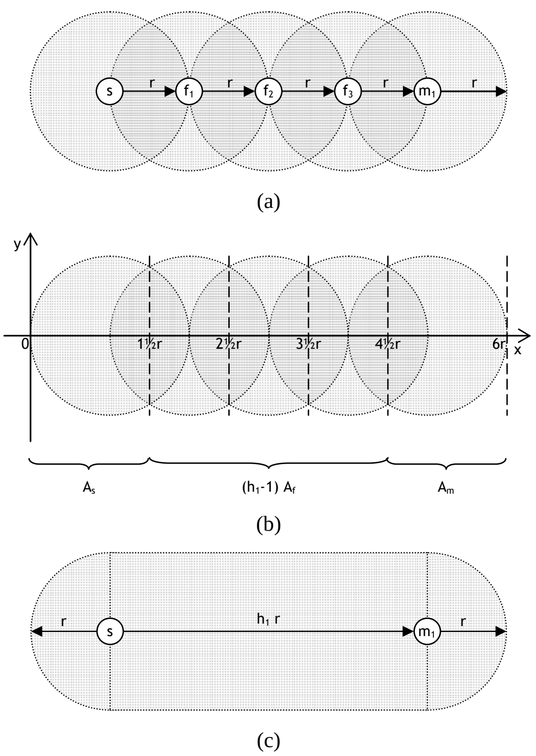


Figure 6. (a) The multicast mesh; (b) its coverage area $A_{M,1}$ and (c) its approximation $A'_{M,1}$ after the first receiver joins the session.

Considering the general case, the number of the intermediate nodes is 1 less than h_1, which is equal to the number of the intermediate areas. Thus, the coverage area of the multicast mesh after the first receiver denoted by $A_{M,1}$ is the superposition of the areas covered by the session server, the intermediate nodes and the receiver:

$$A_{M,1} = A_s + (h_1 - 1)A_f + A_m \tag{6}$$

In order to formulate the integrals and compute the areas A_s, A_f, A_m, their integration boundaries have to be calculated by solving the equations of each pair of intersecting circles for $x = x_h$, where $1 \le h \le h_1$. For instance, the partial area around s to be included to $A_{M,1}$ as A_s is the integral of a circle function over the interval $[0, x_1]$:

$$A_s = 2\int_0^{\frac{3}{2}r} \sqrt{r^2 - (x-r)^2}\,dx \tag{7}$$

Similarly, the partial area around a forwarding node, A_f, can be formulated as the integral of the same circle function over the interval $[x_{h-1}, x_h]$:

$$A_f = 2\int_{\left(h-\frac{1}{2}\right)r}^{\left(h+\frac{1}{2}\right)r} \sqrt{r^2 - (x-hr)^2}\,dx \text{, for } 2 \le h \le h_1. \tag{8}$$

However, since the circles are identical, the calculation of the individual forwarding areas A_f can be made easier by using the same integral as A_s with shifted intervals. Thus:

$$A_f = 4\int_{\frac{1}{2}r}^{r} \sqrt{r^2 - (x-r)^2}\,dx \tag{9}$$

Finally, for the case where $h = h_1$, the area A_m is identical to the area A_s. Thus, the formulation of the total area $A_{M,1}$ given in Equation 6 can be simplified as follows:

$$A_{M,1} = 2A_s + (h_1 - 1)A_f \tag{10}$$

Using Equation 7 and Equation 8 in Equation 6, the integral can be solved as follows:

$$A_{M,1} = 4\left\{\int_0^{\frac{3}{2}r} \sqrt{r^2 - (x-r)^2}\,dx + (h_1 - 1)\int_{\frac{1}{2}r}^{r} \sqrt{r^2 - (x-r)^2}\,dx\right\}$$

$$A_{M,1} = r^2\left\{h_1\left(\frac{\pi}{3} + \frac{\sqrt{3}}{2}\right) + \pi\right\} \tag{11}$$

The probability that the second receiver is within this area, which is also the probability that it can reach the multicast mesh in one hop, is:

$$P\{H_2 = 1\} = \frac{A_{M,1}}{\pi R^2}$$

$$P\{H_2 = 1\} = \frac{r^2}{R^2}\left\{1 + h_1\left(\frac{1}{3} + \frac{\sqrt{3}}{2\pi}\right)\right\} \tag{12}$$

However, it should be noted that Equation 12 only holds as long as $A_{M,1}$ is not larger than the network area.

Otherwise, $P\{H_2 = 1\}$ is equal to 1.

Using Equation 1, the relation of this result to the one-hop probability of the first receiver can be shown as:

$$P\{H_2 = 1\} = P\{H_1 = 1\} + \frac{r^2}{R^2} h_1 \left(\frac{1}{3} + \frac{\sqrt{3}}{2\pi} \right) \quad (13)$$

A second special case should be considered separately, where γ is equal to 1. In this case, $P\{H_1 = 1\}$ as well as $P\{H_2 = 1\}$ are equal to 1 since the transmission ranges cover the whole network. As a result of this, $P\{H_l = 1\}$ equals 1 for all receivers m_l.

Although the coverage area of the multicast mesh following the joining of the first receiver is computed exactly as given by Equation 11, an approximation is provided to simplify subsequent calculations, which is an upper bound to $A_{M,1}$. Thus, the area can be approximated as shown in Figure 6(c) and formulates as follows:

$$A'_{M,1} = r^2 (\pi + 2 h_1) \quad (14)$$

The upper bound of the probability that the second receiver is in this area becomes:

$$P'\{H_2 = 1\} = \frac{r^2}{R^2} \left(1 + \frac{2 h_1}{\pi} \right) \quad (15)$$

The results for $P\{H_2 = 1\}$ and $P'\{H_2 = 1\}$ show that the probability of reaching the session in one hop increases for the second receiver when compared to the first.

$$P'\{H_2 = 1\} = P\{H_1 = 1\} + \frac{r^2}{R^2} \frac{2 h_1}{\pi} \quad (16)$$

This result also provides a looser upper bound to $P\{H_2 = 1\}$ since the distance between the multicast nodes are assumed to be r, the maximum value possible.

The fact that MAQM favours paths with a minimum number of hops between the source and the destination necessitates that the sum of the Euclidian distances d_{ij} between three consecutive nodes on the multicast mesh has the transmission range r as the lower bound. If three nodes on the mesh were placed closer than r, the first node would be able to by-pass the second one and reach the third node directly. In other words, the minimum distance between two consecutive nodes on the mesh is the half of r, the maximum distance is r, and the average distance, r_{avg}, is between these two extremes. Thus, the analysis can be generalised by replacing r by r_{avg} in the equations. Without loss of generality, we continue our analysis using r.

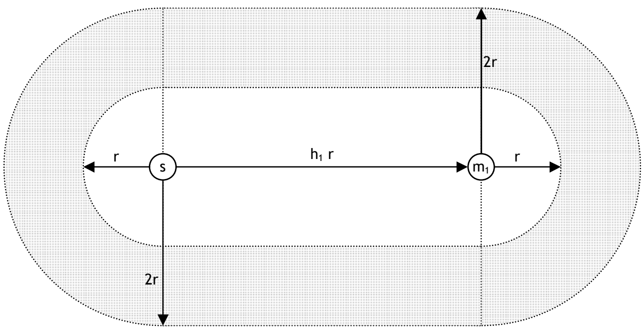


Figure 7. The approximation $A'_{M,2}$ to the two-hop neighbourhood of the multicast mesh.

Having found the coverage area of the one-hop neighbourhood of the multicast mesh, the same approximation can be used to find its two-hop neighbourhood, which is illustrated in Figure 7. These approximations help the derivation of the probabilities $P'\{H_2 = h\}$ for the general case.

The area of the shaded region shown in Figure 7 is:

$$A'_{M,2} = 2 r^2 (2\pi + 2 h_1) - A'_{M,1} \quad (17)$$

Hence, the area of the h-hop neighbourhood can be defined as follows:

$$A'_{M,h} = h r^2 (h\pi + 2 h_1) - A'_{M,h-1} \quad (18)$$

The probability for the second receiver to join the multicast mesh in h hops is:

$$P'\{H_2 = h\} = \frac{A'_{M,h}}{\pi R^2}$$

$$P'\{H_2 = h\} = \frac{r^2}{R^2} \left[(2h - 1) + \frac{2 h_1}{\pi} \right] \quad (19)$$

Similar to Equation 12, Equation 19 only holds as long as $A'_{M,h}$ does not exceed the network area. On the other hand, there is a maximum value that h can take such that $A'_{M,h}$ is not larger than the network, since $A'_{M,h}$ is a function of both h and h_1, whereas h and h_1 are limited by Equation 4 and Definition 4, respectively.

Thus, the h-hop join probabilities for the second session member are:

$$P'\{H_2 = h\} = P\{H_1 = h\} + \frac{r^2}{R^2} \frac{2 h_1}{\pi}, \text{ for } h \le \gamma \quad (20)$$

Similar to the results presented in Equation 16, this result provides an upper bound to $P\{H_2 = h\}$ since the distance between the multicast nodes are assumed to be r.

The results for the h-hop join probabilities of the second receiver can be generalised for the subsequent members of the session. Since the coverage area of the multicast mesh is an increasing function of the number of current receivers, any new member makes it easier for the next join request to reach the multicast mesh in fewer hops. Using this relation, the h-hop join probabilities of the subsequent receivers can be approximated by $P\{H_2 = h\}$.

4.3. Overhead of a Single Join Process

The preceding sections analyse the behaviour of the first and second receivers in a session. With the aid of some simplifications, it is also possible to approximate the behaviour of the subsequent receivers. Given these approximations for the hop count of a new receiver to join the multicast mesh, it is possible to compute the overhead incurred, if the number of the nodes involved in the process at each hop can be determined.

When a session is initiated, it is announced by the server throughout the network. As a result of the nature of the wireless medium, the session initiation messages propagate in the form of an expanding ring. Each node that is informed of the session for the first time forwards these packets only once, which guarantees the downstream flow of the information. The announcement is refreshed periodically by session update packets, which propagate following the same rules. Thus, the expanding ring structure, which groups the nodes according to their distance to the server in terms of hop count, remains intact throughout the session. A join request propagates pretty much in the same fashion as a session initiation, in the form of an expanding ring centred at the node which originates the request. By definitions of MAQM, only those nodes which are aware of the session can satisfy its QoS requirements and are in a ring which is closer to the session server than the requester take the message into consideration. These nodes forward it further upstream towards the session server. The ratio of these nodes in the network can be computed by finding the size of the intersecting areas of the two expanding rings that belong to the requesting and the serving nodes.

An example case is illustrated in Figure 8, where the first requesting node m_1 is just outside the four-hop neighbourhood, or the packet propagation wave $w_{s,4}$ as it is labelled in the figure, of the server s. In this case, the number of nodes that become involved in the request-reply-reserve process can be found by calculating the sum of the areas A_1, A_2, A_3, A_4 and multiplying it with the node density, which is the division of the total number of nodes by the whole network. Following the example of Figure 8, the node m_1, which has a distance slightly greater than 4 r to the server s, finds itself in $w_{s,5}$. Thus, the join request has to propagate five hops to reach the server. This means that there are four groups of nodes between the requester and the server, which forward the request upstream. The first group to process the request consists of those nodes that are within the intersection of $w_{m,1}$ and $w_{s,4}$. These nodes are one-hop closer to the server than the requester. The second, third and fourth groups involved are formed similarly. They are shown in Figure 9(a). In order to find the total number of nodes involved in the join process, the sum of the areas covering these four groups of nodes, namely A_1, A_2, A_3 and A_4, must be calculated. Figure 9(b) shows the intersecting areas to be calculated with their x_h values along the axis.

In order to generalise the case for the join operation, the total area size has to be calculated as the sum of the areas covering all the affected intermediate nodes at each hop:

$$A_J = \sum_{h=1}^{h_1-1} A_h \tag{21}$$

where h_1 is the number of hops from the receiver m_1 to the server s. In other words, the number of intermediate regions between m_1 and s is 1 less than that of hops between them.

To formulate the integrals and compute the areas A_1, A_2, A_3, A_4, the integration boundaries are calculated by solving the equations of each pair of intersecting circles for x_h as:

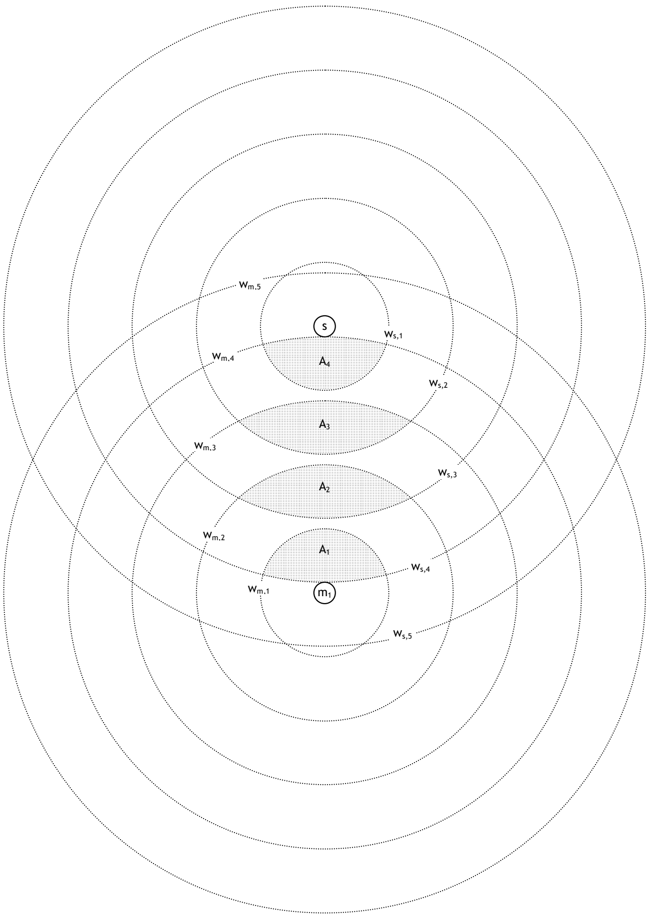


Figure 8. The areas involved in the propagation of a join request from the requester m_1 towards the server s.

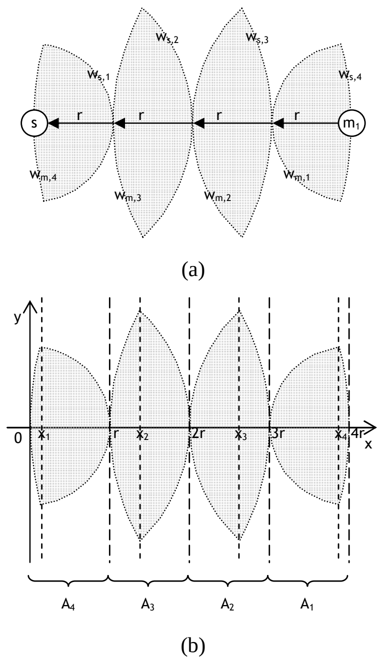


Figure 9. (a) The areas containing the nodes involved in the join process and (b) their integral boundaries.

$$h^2r^2 - x_h^2 = (h_1 - h)^2 r^2 - (x_h - [h_1 - 1]r)^2$$

$$x_h = \frac{r}{2}\frac{2h_1(h-1)+1}{h_1 - 1}, \text{ for } 1 \le h \le h_1\text{-1} \qquad (22)$$

Thus, the x_h value of the intersection is a function of the current hop h. The total area involved in the join process, A_J, which is the sum of all the integrals A_h, is:

$$A_J = 2\sum_{h=1}^{h_1-1}\left\{\begin{array}{c}\int_{(h-1)r}^{x_h}\sqrt{(h_1-h)^2r^2-(x-[h_1-1]r)^2}\,dx + \\ \int_{x_h}^{hr}\sqrt{h^2r^2-x^2}\,dx\end{array}\right\} \qquad (23)$$

where $1 \le h \le h_1$-1 and x_h is determined for each term using the function given in Equation 22.

Using ν, the total number of nodes in the ad hoc network, ν_J, the number of nodes within the area A_J involved in the join request of m_1, can be found:

$$\nu_J = \nu \frac{A_J}{\pi R^2} \qquad (24)$$

It is known that the join process consists of the request, reply and reserve phases in MAQM. The request and reply packets are forwarded by ν_J nodes towards the server as explained above, whereas the reservation packets in the last phase are aimed at exactly one selected upstream node. Thus, the total number of control messages μ_J processed by the intermediate nodes is:

$$\mu_J = 2\nu_J + h_1 - 1 \qquad (25)$$

Using this formula with the expected number of hops a receiver needs to join a multicast mesh, it is possible to estimate the control overhead of a typical join operation or the overhead per session, per member, per time unit.

With the help of the symmetry of the shape along the axis crossing the midpoint of the $s - m_1$ line at $(h_1 - 1)r/2$ and applying substitution techniques, A_J can be exactly determined. However, since both the sum of integrals as well as the intervals of each integral also depend on h, it is preferable to use one of the approximations in Figure 10 for the calculation of the area involved in the join process.

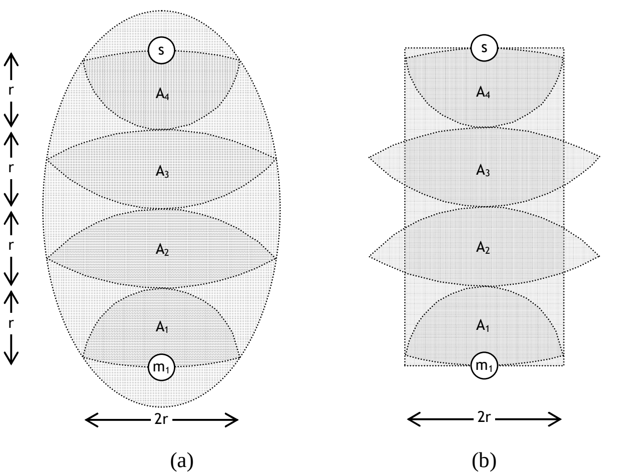


Figure 10. Two approximations, (a) A'_J and (b) A''_J, to the area involved in the join process.

It can be argued that for a small number of hops, the rectangular approximation is more appropriate. However, the ellipse is the only shape that covers all of the partial areas regardless of h_1, which makes it the preferred approximation to find an upper bound for the total area. By definition, an ellipse is the set of points in a plane whose distances from two fixed points in the plane have a constant sum. Since the sum of the radii of the intersecting expanding rings centred at s and m_1 is always h_1, it is obvious that the ellipse always covers A_1, A_2, A_3, and A_4.

The area of the ellipse can be formulated as follows:

$$A'_J = \frac{\pi r^2}{4} h_1 \sqrt{2h_1 - 1} \qquad (26)$$

The area of the rectangle can be formulated as follows:

$$A''_J = 2(h_1 - 1)r^2 \qquad (27)$$

Selecting the ellipse as the approximation to the area involved in the join request is also useful for smoothing away the decisional errors made by some of the nodes. Since the session update messages have a certain period, it is possible that there are nodes that react to join requests although they should not. These nodes may have lost their connection to the session or moved away from the location where they have been able to support it. In other words, a topological change in the network, which is a result of node mobility as well as the properties of the wireless medium, may cause some of the nodes that are actually outside the involved area to take part falsely in the process. Thus, an ellipse reaching from the requester to the session server is a logical approximation to represent the propagation of the control messages between these two nodes.

4.4. Interpretation of Results

This section has presented an analysis of the control overhead during the join process of MAQM. The propagation of a join request is examined in order to find an average value for the number of intermediate nodes involved and, thus, the number of packets propagating in the network during a typical join process. The results presented in Section 4.1 and Section 4.2 show that, after the first member of a session connects to the server, the next join request can be fulfilled in fewer hops, requiring a lower control overhead. Moreover, the average hop count required by the second member decreases as the path between the first member and the server becomes longer. This result confirms that the increase in the number of intermediate nodes leads to a larger aggregate coverage area, which makes it easier for the second member to reach the mesh. We can further argue that the trend of decreasing average values continues for the subsequent members of the session and the multicast mesh gets probabilistically closer for each new member. Hence, the results achieved for the second receiver can be used as the worst-case value for all nodes. Combining these results, more general estimates can be

Table 2. QoS requirements of the application classes.

QoS Class	Bandwidth Requirement	Average Duration	Delay Limit	Application Type
1	128 Kbps	300 s	10 ms	Voice conversation
2	256 Kbps	900 s	50 ms	Streaming music
3	512 Kbps	600 s	10 ms	Video conference
4	2 Mbps	1 200 s	50 ms	Streaming video

obtained such as the average or worst-case control overhead per session, per member, or per unit time throughout the network. These estimates can be used to adjust some MAQM parameters such as session update and greeting message intervals to improve resource efficiency.

The analytical results presented in this section can be used to derive the number of nodes involved in a join process with the help of Equations 24, 26 and 27. This part of the analysis is verified through simulation in Section 5.3. Once this is done, the number of processed control packets is easy to derive via Equation 25.

As mentioned at the beginning of this section, the registration of some receivers by extra forwarders during the multicast data streaming is not taken into consideration since the tree-to-mesh evolution presented in Section 3.5 does not affect the analysis. It can be seen easily that the analysis is still valid with the mesh option since the extra forwarders to be registered are already members of their respective sessions. Therefore, the join process of a subsequent receiver remains the same. In other words, the tree-to-mesh evolution does not affect the results of the analysis since only new wireless links, and not any new nodes, are added to the multicast tree during this operation. Since the coverage area of the multicast group remains the same, the analysis of the join process covers both the initial multicast tree as well as the resulting multicast mesh.

5. Performance Evaluation

The objective of this section is to test the efficiency of MAQM in providing multicast users with QoS and satisfying the service requirements of multimedia applications. The simulations are conducted using OPNET Modeler 11.5 Educational Version and Wireless Module [36]. Simulations are repeated 10 times for each data point and results are aggregated with a 95 % confidence interval in a multicast scenario with four QoS classes as defined in Table 2 to represent a set of applications coexisting in the system. To comply with the bandwidth requirements and delay tolerance characteristics given as part of these sample QoS definitions, nodes are restricted to certain minimum bandwidth and maximum hop count regulations. Hence, a node is allowed to join a session only if it can find a path to the server with more bandwidth available than the allowed minimum and less hops away than the allowed maximum. There are no limits to the size of the multicast groups.

The effect of mobility is observed under the random waypoint mobility model with uniformly distributed node speeds and pause times representing pedestrian mobility. In contrast to previous performance evaluations in the research literature, which limit their simulations to a few minutes, one hour of network lifetime has been simulated to get a realistic impression of the aggregated behaviour of multiple multicast sessions being served simultaneously in a distributed manner. Background data traffic co-exists with multicast data traffic in order to observe its effects on the performance. The parameters of the mobility model and other simulation settings are given in Table 3. 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 network infrastructure is not available and nodes move around with walking or running speeds.

Table 3. Simulation parameters for the evaluation.

Parameter Description	Value
Background traffic inactivity period	300 s (exponential)
Background traffic file size	2 MB (lognormal)
Greeting message interval	10 s
Maximum link bandwidth	10 Mbps
Mobility model	Random waypoint
Node pause time	10 – 40 s (uniform)
Node speed	1 – 4 m/s (uniform)
Multicast inactivity period	100 s (exponential)
Network population	100 nodes
Session update message interval	15 s
Simulation duration	1 h
Wireless transmission range	250 m

The non-QoS protocol developed for comparison purposes resembles basically a modified version of MAODV [9]. However, MAODV utilises the information collected during the unicast route discovery, which is not implemented in the non-QoS protocol developed for the simulations to achieve fair comparison conditions. MAODV maintains sequence numbers for multicast groups, which are updated by the group leaders, to ensure that the most recent route to the multicast group is used. Like MAQM, the non-QoS protocol supports multiple sessions as well as multiple service classes simultaneously. However, it does not make any intelligent decisions based on QoS availability when responding to join requests.

Two sets of simulations are conducted with these common parameters. The first set examines the effect of network density on MAQM. In this set, the average number of neighbours within a node's transmission range is the variable. The second set, whereby the percentage of the sessions which belong to the heaviest service class is the variable, aims to test the effect of the changes in multicast traffic load on MAQM. In this set, the other classes share the remaining occurrence probability equally.

5.1. Satisfaction of Session Members

The success of a QoS multicast routing system depends primarily on the satisfaction of its members. In this regard, the most important criterion for the QoS-related multicast routing decisions made by MAQM is the improvement in the ratio of session members satisfied by the perceived quality of their applications. It is one of MAQM's main concerns that network resources are not excessively utilised to avoid possible collisions, packet loss and delay due to overload and keep the QoS conditions at a satisfactory level. Once accepted to a session, the QoS status perceived by the nodes during the course of the session is vital. Thus, it is necessary to observe changes in member-level QoS. The member QoS sustainability ratio Q_{Member} is defined to evaluate this aspect of MAQM and formulated as follows:

$$Q_{Member} = 1 - \frac{d}{a} \tag{28}$$

where d is the number of members dropped off a session due to insufficient QoS and a represents the number of nodes accepted to sessions as receivers. The decision on the sustainability of QoS is based on a combination of various other QoS metrics such as the end-to-end delay, interarrival time and loss rate of the data packets. Thus, members are dropped when the flow of data they receive cannot meet the QoS standards required by their multicast session. Equation 28 gives the percentage of members which are served by the ad hoc network with acceptable QoS during their entire session membership. The member QoS sustainability ratio is an important criterion for the evaluation of member satisfaction since it is also a measure of the percentage of members experiencing severe delay and loss problems due to allocations exceeding the resource limits of the network.

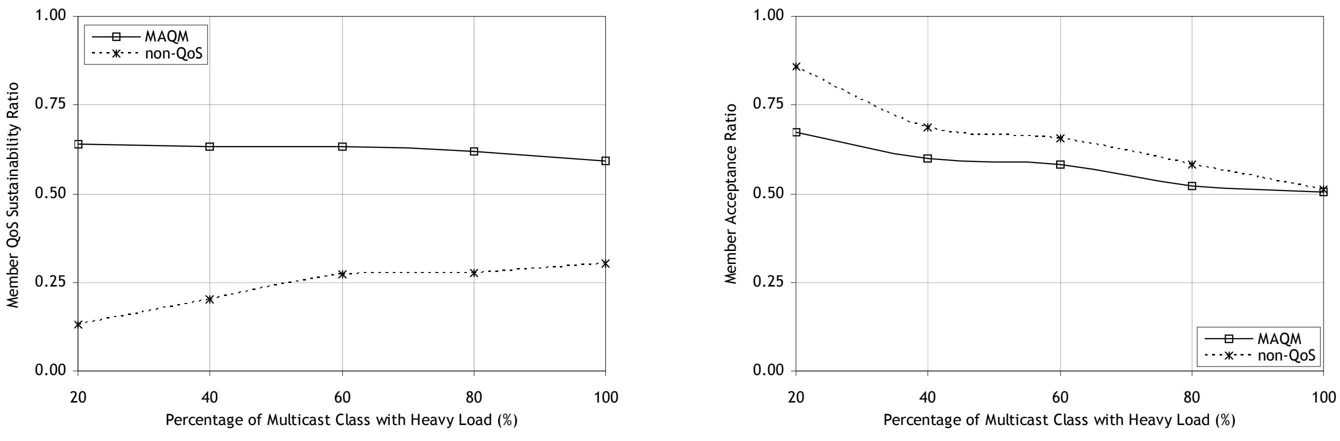
The success rate of member satisfaction is an important criterion for the performance of a multicast routing protocol providing QoS. On the other hand, it has to be taken into account that the member satisfaction achieved by the prevention of overload has an effect on the system, which can be observed by the percentage of users that are admitted to the multicast sessions. An efficient QoS multicast protocol should not allow its user admission rate to drop unacceptably as a result of the application of QoS restrictions. In other words, the majority of the users who wish to join a multicast session should still be admitted even with QoS limitations. Thus, the member acceptance ratio A_{Member} is formulated as follows:

$$A_{Member} = \frac{a}{g} \tag{29}$$

where g is the total number of join requests issued by all ad hoc nodes. The ratio reflects the success rate of MAQM in accepting a node’s request to join a session. The member acceptance ratio is an important performance metric. Thus, it is essential that a QoS-aware multicast protocol maintains a good balance between these two aspects of satisfaction.

The two performance metrics defined above to evaluate the member satisfaction, should rather be interpreted together in order to see their relation. The effect of traffic load is evaluated by increasing the ratio of initiated sessions with higher QoS requirements in the network. A major conclusion drawn from the simulation results presented in this section is that there is a trade-off between member acceptance and sustainability of QoS. Thus, MAQM lets one of them degrade gracefully in order to maintain the other at an acceptable level when necessary. The logic behind these decisions is explained below.

Figure 11 compares the member satisfaction performance of MAQM to a non-QoS scheme in terms of (a) member QoS sustainability and (b) the number of accepted session members as the ratio of multicast sessions that belong to the class with higher QoS requirements increases. It can be seen from the figure that MAQM is able to sustain the membership QoS for a significant portion of the members once it accepts them to a session, whereas the non-QoS protocol can only provide poor QoS conditions to its users, mainly due to the fact that it accepts too many join requests without considering the resource limitations of the network. On the other hand, MAQM maintains the QoS level of its accepted members at the cost of decreasing its member acceptance ratio. It keeps its nodes up-to-date regarding the QoS conditions in the network and the status of the existing sessions. MAQM nodes do not accept new requests if they cannot afford the required bandwidth and hop count requirements. It can also use more up-to-date resource allocation information when there are fewer simultaneous requests. Thus, not all requests are granted an acceptance and the member acceptance ratio is generally lower than a non-QoS protocol. However, it should be noted that the increase in the QoS sustainability



(a) Member QoS sustainability ratio as a function of multicast traffic load

(b) Member acceptance ratio as a function of multicast traffic load

Figure 11. Comparison of MAQM to a non-QoS scheme with regard to satisfaction of session members as multicast traffic load increases.

performance of the non-QoS protocol is the result of the decrease in its member acceptance ratio due to loss of control messages under heavy data traffic. MAQM is still able to achieve an acceptance ratio close to its competitor.

Another important aspect of the results presented in Figure 11 is the fact that the non-QoS protocol cannot achieve the QoS sustainability rate of MAQM even though its member acceptance ratio is very close to MAQM for higher rates of heavy-class multicast traffic. This is a clear indication that MAQM is more than just admission control. A sustainable QoS rate close to that of MAQM cannot be achieved merely by accepting join requests randomly at a rate close to that of MAQM. MAQM has other important features such as a QoS-controlled join process, resource allocation and hop count limitation, leading to a more balanced network load and increasing the ratio of member satisfaction. While the application of QoS restrictions causes more users to be rejected, the lack of these restrictions yields to performance degradation in the network. Without a policy to manage network resources effectively, users experience difficulties in getting any service as the resource requirements increase.

5.2. Effects on Network Delay and Control Overhead

As stated above, MAQM decides on the sustainability of a membership based on a combination of various QoS metrics such as the loss ratio, average end-to-end delay and interarrival time of the data packets. This is a necessary countermeasure in order to protect the network from overall performance degradation at the members' level, which is shown in the preceding section. In this section, a more thorough examination of end-to-end delay and interarrival time is provided in order to show an example of MAQM's effect on two of the aforementioned QoS metrics.

It is inevitable that the computational overhead of a routing protocol increases with its complexity. However, it is possible to keep this overhead at an acceptable level while adding QoS functionality to a protocol, especially in order to deal with the effects of mobility, the changes in topology and the issues of scalability. Thus, the member control overhead C_{Member} is formulated as follows:

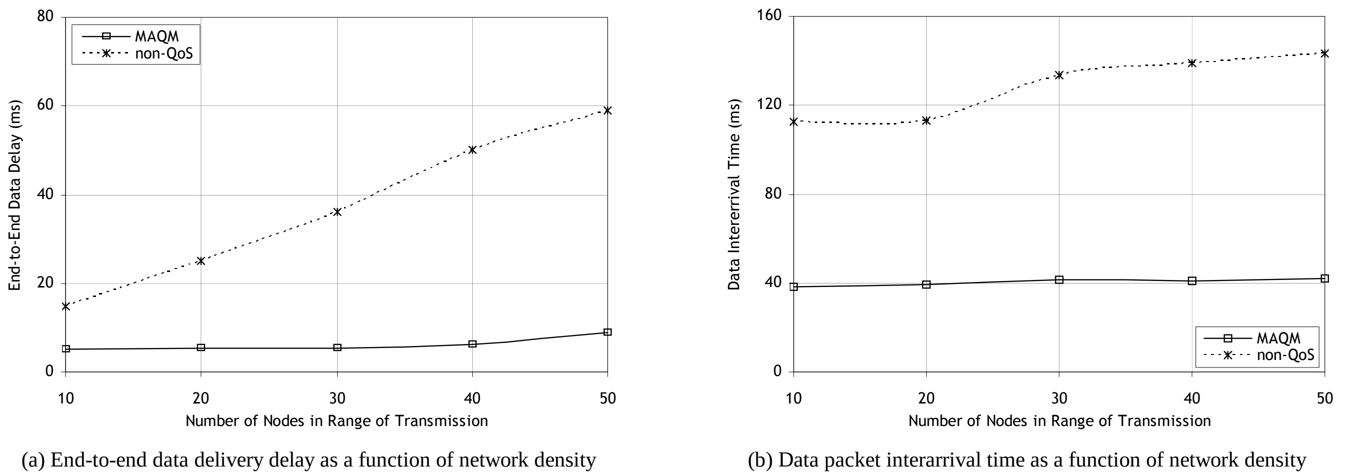
$$C_{Member} = \frac{c}{z + f + a} \qquad (30)$$

where c represents the total number of multicast control packets received and processed by the nodes of the ad hoc network, z is the number of session servers and f is the number of forwarders. The sum of z, f and a gives the total number of active nodes in the network. An active node is a session member participating in at least one multicast session as a server, forwarder or receiver. Thus, the division gives the number of control packets per multicast member to maintain the MAQM system. The nature of ad hoc networks requires that such a protocol works with an additional overhead as little as possible. Therefore, it is necessary that the control overhead incurred by MAQM is evaluated to have an idea on its effects on the network load.

Figure 12 shows the performance of MAQM and the non-QoS protocol with regard to their (a) end-to-end data delay and (b) data interarrival time. The main reason for end-to-end delay is contention, whereas the average interarrival time increases due to collisions. Both happen much rarer in MAQM as a result of its ability to reserve resources and balance network load. There is only a slight increase in the end-to-end delay of MAQM as the network becomes denser and a similar behaviour is observed in its interarrival time. The averages of the non-QoS protocol are higher than acceptable and increase drastically with network density. These results show that MAQM is able to deliver data packets in a streaming fashion as required by multimedia applications, providing stable QoS conditions.

Figure 13 compares the member control overhead of MAQM to the non-QoS protocol (a) as the network density and (b) the ratio of multicast sessions that belong to the class with higher QoS requirements increases. The unit of control overhead is defined as the number of packets received and processed by a session member per second.

As shown in Figure 13(a), the average overhead increases in a denser network, mainly due to more join requests and replies forwarded as a result of higher



(a) End-to-end data delivery delay as a function of network density

(b) Data packet interarrival time as a function of network density

Figure 12. Comparison of MAQM to a non-QoS scheme with regard to effects on network delay as network density increases.

connectivity. Since the non-QoS protocol forwards these types of messages without QoS considerations anyway, the increased connectivity does not affect its control overhead as much as MAQM. Another reason for MAQM's increasing overhead is its member dropping process due to lack of acceptable QoS, which triggers subsequent actions at other session members both upstream as well as downstream. On the other hand, MAQM eliminates infeasible join request at their sources and deals with less membership operations in general. Moreover, by rejecting some join requests, MAQM cuts further communication with those nodes at an early stage of the process. Finally, MAQM uses additional forwarders during data streaming, which increases the robustness of the multicast graphs and helps the protocol experience fewer session losses. These features save MAQM from a higher control overhead.

As shown in Figure 13(b), the overhead of both protocols grows slightly as the ratio of heavy class sessions is increased. The main reason for this behaviour common to both protocols is the fact that they reject more join requests, which yields to new requests and replies that are subsequently forwarded. MAQM, on the other hand, facilitates additional control messages for status updates regarding its sessions and extra forwarders which are notified during multicast data flow to improve robustness.

On the other hand, there is an obvious difference in the scale of overhead in MAQM between both figures. This is due to the fact that Figure 13(b) displays the results from a scenario with relatively heavier average traffic load than Figure 13(a). As a consequence, there is relatively less available bandwith, which yields to a fewer number of QOS-related communication, such as the propagation of session information, join requests and all related messages.

Although the control overhead incurred by MAQM is generally higher than the non-QoS protocol and the number of control packets per member grows as the network density increases, it is worth mentioning that this overhead is actually very small when compared to the multimedia data traffic. Considering average bandwidth requirements and session durations defined by the scenario in Table 2, the average data traffic per session is on the order of megabits per second. On the other hand, the size of the largest MAQM control packet is around 200 bits, including various lower layer headers. MAQM does not exchange the information on neighbours, members and sessions in the form of long lists. Therefore, it does not need to use variable-size control packets. The worst-case average control traffic per member is on the order of 0.5 kilobits per second, which is the rate experienced in a highly dense network. Thus, the increased overhead of MAQM is still reasonable considering the fact that it achieves much higher member satisfaction for the users in the ad hoc network.

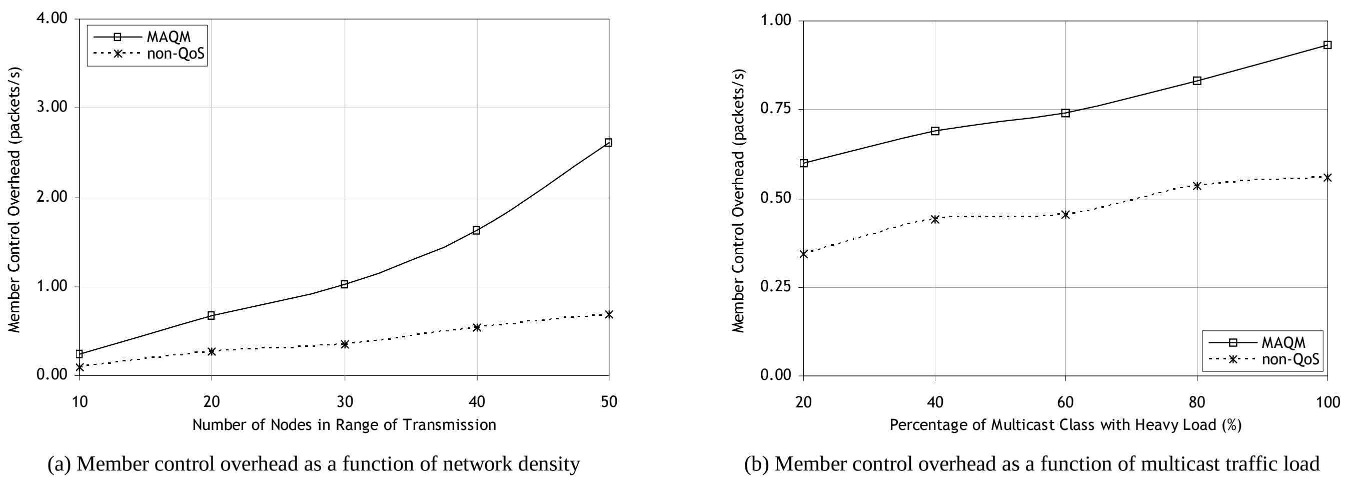
In this section, so far the control overhead experienced by MAQM nodes is observed in a general context, whereby it is evaluated only quantitatively, in other words, without classification. In Section 4, the control overhead is analysed thoroughly with particular emphasis on the session join process, which is the most interactive part of the protocol. Therefore, a deeper look is provided in the next section.

5.3. Validation of the Overhead Analysis

A separate set of simulations are conducted using OPNET Modeler 11.5 Educational Version and Wireless Module in order to validate the analytical results, where the simulations are repeated 10 times for each data point and results are aggregated with a 95 % confidence interval. The nodes are placed randomly in a circular area. There is only one server placed at the centre of the circle. Mobility is omitted. Other simulation settings are presented in Table 4.

Table 4. Simulation settings for the analysis of the join process.

R/r	Number of Receivers	Network Radius	Transmission Range
2	25	500 m	250 m
3	36	600 m	200 m
4	49	700 m	175 m
5	64	800 m	160 m
6	81	900 m	150 m
7	100	1 000 m	145 m



(a) Member control overhead as a function of network density

(b) Member control overhead as a function of multicast traffic load

Figure 13. Comparison of MAQM to a non-QoS scheme with regard to effects on control overhead.

In order to achieve constant node density, the network population is kept proportional to R^2.

Figure 14(a) compares the number of nodes involved in the join process of the first session member experienced in the simulation with the values computed analytically. As mentioned in Section 4, the analysis is based on the expected values of the hop distance between the candidate and the server and provides an approximation to the number of intermediate nodes in the propagation area of the join request as given in Equation 27. The simulation results follow the trend suggested by the analysis.

Figure 14(b) makes the same comparison for the second join attempt after the multicast tree between the server and the first member is initialised and can be evolved into a mesh. This time, Equation 26 is used to approximate the size of the propagation area with slightly looser bounds. This way, the possible divergence at the boundaries of the area due to the mobility of multiple nodes building the mesh is covered better. It can be seen that the expected values of the analysis as well as the results achieved by simulation are below the averages of the first member. Thus, it is shown both analytically and experimentally that the overhead of a join attempt by subsequent candidates decreases as more nodes become session members. On the other hand, the simulation results are closer to the analytical results and follow the same trend which suggests that the number of nodes taking part in the join process increases gracefully as the size of the network grows.

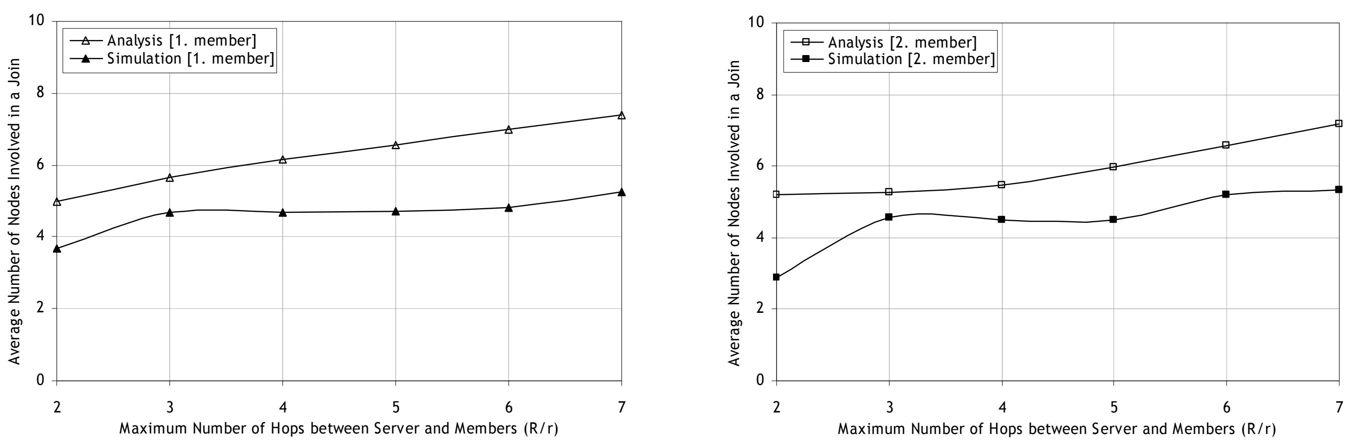
According to both the analysis as well as the simulation results, MAQM provides a session and membership management system to its users, whereby each new session member can join the multicast tree with an acceptable overhead, which degrades gracefully for each session.

6. Conclusion

The multimedia content shared over communication media today makes QoS-related, resource-efficient routing strategies very important. At the same time, mobile ad hoc networks are becoming increasingly popular since they provide the user with the ability to communicate anytime, anywhere. Group-oriented applications in these networks necessitate the efficient utilisation of resources. Multicast is a promising technique, which can achieve this efficiency by facilitating the inherent broadcast capability of the wireless medium. MAQM is a multicast protocol for mobile ad hoc networks, which provides QoS-aware strategies for admission control and resource allocation.

MAQM introduces novel ideas to ad hoc multicast routing. It defines the bandwidth requirement of a session as a continuous flow of multimedia data, which we call a virtual tunnel of bandwidth, and makes accurate decisions on resource availability. It checks the bandwidth availability within each node's neighbourhood based on previous reservations, and ensures that updated QoS information is used to select the routes meeting the service requirements of a session. MAQM is also able to evolve the initial multicast tree into a mesh during data flow. A node connects to the existing multicast graph via a single forwarding member but is allowed to register itself with additional forwarders if it starts receiving multicast data from them. This operation increases robustness since a node does not only depend on a single predecessor for its connection to the multicast session. It is also important that robustness is achieved with a small overhead. Our analysis of the session joining process shows that this overhead is actually acceptable.

In addition to regular performance metrics such as the multicast success rate and the control overhead, the concept of QoS sustainability is introduced to evaluate MAQM with regard to members with insufficient perceived QoS, which has a direct impact on service satisfaction. MAQM's performance is evaluated with regard to these metrics under a realistic network scenario, where multiple QoS classes are supported with no restrictions on the number of simultaneous sessions or members. Background data traffic is incorporated in the scenario along with multicast data traffic in order to observe its effect on the performance. Simulation results show that, by applying QoS restrictions to the ad hoc network, MAQM significantly improves the



(a) Number of nodes involved in the join process of the first receiver

(b) Number of nodes involved in the join process of the second receiver

Figure 14. Comparison of the analysis of the tree-to-mesh evolution process to the simulation results.

multicast sustainability. Without a QoS policy, members experience difficulties in getting the service they demand as the traffic load grows and network density increases.

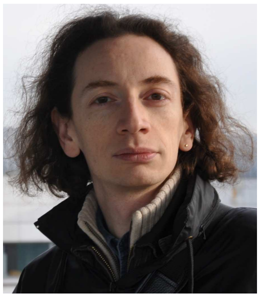
It is possible to improve MAQM further with the help of additional information collected by the nodes in a distributed manner and shared among the neighbours in the network. Nodes can measure their queue sizes and estimate their average queuing delays. They can also measure their processing delays and derive a relation to the number of sessions being processed by them. More sophisticated admission, reservation and routing decisions can be made using these shared observations and the results provided by the analysis of the control overhead. MAQM prevents the excessive allocation of bandwidth and helps the nodes experience less contention, which also affects delay. Since delay is mainly the sum of contention and transmission times, extra information on queuing and processing delays can be valuable for MAQM to select paths of lower delay. Nevertheless, MAQM proves that QoS support is essential for multimedia communication in mobile ad hoc networks.

References

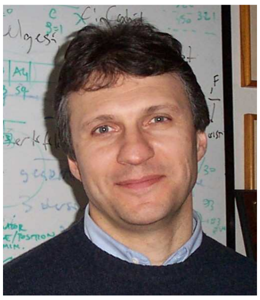
1. Reddy, T.B., I. Karthigeyan, B.S. Manoj, C.S.R. Murthy, “Quality of Service Provisioning in Ad Hoc Wireless Networks: A Survey of Issues and Solutions”, *Ad Hoc Networks,* Vol. 4, No. 1, pp. 83-124, January 2006.
2. Murthy, C.S.R., B.S. Manoj, *Ad Hoc Wireless Networks – Architectures and Protocols,* Prentice Hall PTR, 2004.
3. Bür, K. C. Ersoy, “Ad Hoc Quality of Service Multicast Routing”, *Computer Communications*, Vol. 29, No. 1, pp. 136-148, December 2005.
4. Viswanath, K., K. Obraczka, G. Tsudik, “Exploring Mesh and Tree-Based Multicast Routing Protocols for MANETs”, *IEEE Transactions on Mobile Computing*, Vol. 5, No. 1, pp. 28-42, January 2006.
5. Junhai, L., X. Liu, Y. Danxia, “Research on Multicast Routing Protocols for Mobile Ad Hoc Networks”, *Computer Networks*, Vol. 52, No. 5, pp. 988-997, April 2008.
6. Liu, M., R.R. Talpade, A. McAuley, E. Bommaiah, “AMRoute: Ad Hoc Multicast Routing Protocol”, *Center for Satellite and Hybrid Communication Networks Technical Research Report*, University of Maryland, August 1999.
7. Gerla, M., C.-C. Chiang, L. Zhang, “Tree Multicast Strategies in Mobile, Multihop Wireless Networks”, *ACM Mobile Networks and Applications*, Vol. 4, No. 3, pp. 193-207, October 1999.
8. Wu, C.W., Y.C. Tay, C.K. Toh. “Ad Hoc Multicast Routing Protocol Utilizing Increasing Id-Numbers (AMRIS) Functional Specification”, *IETF MANET WG Internet Draft (draft-ietf-manet-amris-spec-00.txt)*, work in progress, November 1998.
9. Royer, E.M., C.E. Perkins, “Multicast Ad Hoc On-Demand Distance Vector (MAODV) Routing,” *IETF MANET WG Internet Draft (draft-ietf-manet-maodv-00.txt)*, work in progress, July 2000.
10. Wang, X., F. Li, S. Ishihara, T. Mizuno, “ A Multicast Routing Algorithm Based on Mobile Multicast Agents in Ad Hoc Networks”, *IEICE Transactions on Communication*, Vol. E84-B, No. 8, pp. 2087-2094, August 2001.
11. Ng, J.M., C.P. Low, H.S. Teo, “On-Demand QoS Multicast Routing and Reservation Protocol for MANETs”, *Proceedings of IEEE PIMRC 2004*, Vol. 4, pp. 2504-2508, Barcelona, Spain, 5-8 September 2004.
12. Shen, C.-C., C. Jaikaeo, “Ad Hoc Multicast Routing Algorithm with Swarm Intelligence”, *ACM Mobile Networks and Applications*, Vol. 10, No. 1-2, pp. 47-59, February 2005.
13. Layuan, L., L. Chunlin, “A QoS Multicast Routing Protocol for Clustering Mobile Ad Hoc Networks”, *Computer Communications*, Vol. 30, No. 7, pp. 1641-1654, May 2007.
14. Sinha, P., R. Sivakumar, V. Bharghavan, “MCEDAR: Multicast Core-Extraction Distributed Ad Hoc Routing”, *Proceedings of IEEE WCNC 1999*, Vol. 3, pp. 1313-1317, New Orleans, USA, 21-24 September 1999.
15. Ozaki, T., J.B. Kim, T. Suda, “Bandwidth-Efficient Multicast Routing for Multihop, Ad Hoc Wireless Networks”, *Proceedings of IEEE Infocom 2001*, Vol. 2, pp. 1182-1191, Alaska, USA, 22-26 April 2001.
16. Ji, L., M.S. Corson, “Explicit Multicasting for Mobile Ad Hoc Networks”, *ACM Mobile Networks and Applications*, Vol. 8, No. 5, pp. 535-549, October 2003.
17. Toh, C.K., G. Guichala, S. Bunchua, “ABAM: On-Demand Associativity-Based Multicast Routing for Ad Hoc Mobile Networks”, *Proceedings of IEEE VTC 2000 Fall*, Vol. 3, pp. 987-993, Boston, USA, 24-28 September 2000.
18. Devarapalli, V., A.A. Selçuk, D. Sidhu, “MZR: A Multicast Protocol for Mobile Ad Hoc Networks”, *IETF MANET WG Internet Draft (draft-vijay-manet-mzr-01.txt)*, work in progress, July 2001.
19. Das, S.K., B.S. Manoj, C.S.R. Murthy, “Weight-Based Multicast Routing Protocol for Ad Hoc Wireless Networks”, *Proceedings of IEEE Globecom 2002*, Vol. 1, pp. 117-121, Taipei, Taiwan, 17-21 November 2002.
20. Sisodia, R.S., I. Karthigeyan, B.S. Manoj, C.S.R. Murthy, “A Preferred Link-Based Multicast Protocol for Wireless Mobile Ad Hoc Networks”, *Proceedings of IEEE ICC 2003*, Vol. 3, pp. 2213-2217, Anchorage, USA, 11-15 May 2003.
21. Wu, H., X. Jia, “QoS multicast routing by using multiple paths/trees in wireless ad hoc networks”, *Ad Hoc Networks*, Vol. 5, No. 5, pp. 600-612, July 2007.
22. Manvi, S.S., M.S. Kakkasageri, “Multicast Routing in Mobile Ad Hoc Networks by Using a Multiagent System”, *Information Sciences*, Vol. 178, No. 6, pp. 1611-1628, March 2008.
23. Chiang, C.-C., M. Gerla, L. Zhang, “Forwarding Group Multicast Protocol (FGMP) for Multihop, Mobile Wireless Networks”, *ACM/Baltzer Journal of Cluster Computing*, Vol. 1, No. 2, pp. 187-196, November 1998.
24. Madruga, E.L., J.J. Garcia-Luna-Aceves, “Scalable Multicasting: The Core-Assisted Mesh Protocol”, *ACM Mobile Networks and Applications*, Vol. 6, No. 2, pp. 151-165, March 2001.
25. Lee, S.J., W. Su, M. Gerla, “On-Demand Multicast Routing Protocol in Multihop Wireless Mobile Networks”, *ACM Mobile Networks and Applications*, Vol. 7, No. 6, pp. 441-453, December 2002.
26. Lee, S., C. Kim, “A new wireless ad hoc multicast routing protocol”, *Computer Networks*, Vol. 38, No.2, pp. 121-135, February 2002.
27. Moustafa, H., H. Labiod, “A Performance Analysis of Source Routing-Based Multicast Protocol (SRMP) Using Different Mobility Models”, *Proceedings of IEEE ICC 2004*, Vol. 7, pp. 4192-4196, Paris, France, 20-24 June 2004.
28. Das, S.K., B.S. Manoj, C.S.R. Murthy, “A Dynamic Core-Based Multicast Routing Protocol for Ad Hoc Wireless Networks”, *Proceedings of ACM MobiHoc 2002*, pp. 24-35, Lausanne, Switzerland, 9-11 June 2002.
29. Luo, J., P.T. Eugster, J.-P. Hubaux, “Probabilistic reliable multicast in ad hoc networks”, *Ad Hoc Networks*, Vol. 2, No. 4, pp. 369-386, October 2004.
30. Vaishampayan, R., J.J. Garcia-Luna-Aceves, “Efficient and Robust Multicast Routing in Mobile Ad Hoc Networks”, *Proceedings of MASS 2004*, pp. 304-313, Fort Lauderdale, USA, 24-27 October 2004.
31. Wang, G., J. Cao, L. Zhang, K.C.C. Chan, J. Wu, “A Novel QoS Multicast Model in Mobile Ad Hoc Networks”, *Proceedings of IEEE IPDPS 2005*, pp. 206b-206b, Denver, USA, 4-8 April 2005.
32. Mnaouer, A.B., L. Chen, C.H. Foh, J.W. Tantra, “OPHMR: An Optimized Polymorphic Hybrid Multicast Routing Protocol for MANET”, *IEEE Transactions on Mobile Computing*, Vol. 5, No. 6, pp. 503-514, May 2007.
33. Tebbe, H., A.J. Kassler, P.M. Ruiz, “QoS-Aware Mesh Construction to Enhance Multicast Routing in Mobile Ad Hoc Networks”, *Proceedings of InterSense 2006*, Nice, France, 30-31 May 2006.
34. Shah, S.H., K. Chen, K. Nahrstedt, “Dynamic Bandwidth Management in Single-Hop Ad Hoc Wireless Networks”, *ACM Mobile Networks and Applications*, Vol. 10, No. 1, pp. 199-217, February 2005.

35. Takagi, H., L. Kleinrock, "Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals", *IEEE Transactions on Communications*, Vol. COM-32, No. 3, pp. 246-257, March 1984.
36. OPNET Technologies Inc., Bethesda, MD, USA, available at http://www.opnet.com, last accessed September 2008.

Biographies



Kaan Bür received his BSc degree in control and computer engineering from Istanbul Technical University in 1995, and his MSc and PhD degrees in computer engineering from Boğaziçi University, Istanbul, in 1998 and 2006, respectively. From 1995 to 2004, he worked as an R&D engineer for the consumer electronics industry. From 2006 to 2007, he was a senior researcher at Faculty of Electrical Engineering and Computer Sciences, Berlin Technical University. From 2008 to 2009, he was a research associate at Department of Electronic and Electrical Engineering, University College London. Since 2009, he is a postdoctoral researcher at Department of Electrical and Information Technology, Lund University. His current research interests include quality of service, next generation networks, wireless sensor and mesh networks, mobile and vehicular ad hoc networks.



Cem Ersoy received his BSc and MSc degrees in electrical engineering from Boğaziçi University, Istanbul, in 1984 and 1986, respectively. He worked as an R&D engineer at NETAS A.S. between 1984 and 1986. He received his PhD in electrical engineering from Polytechnic University, Brooklyn, New York in 1992. Currently, he is a professor at Department of Computer Engineering, Boğaziçi University. His research interests include performance evaluation of communication networks, wireless sensor networks and mobile applications. Dr. Ersoy is a senior member of IEEE and the chairman of IEEE Communications Society Turkey Chapter.