Admission control for multicast routing with quality of service in ad hoc networks

Bür, Kaan; Ersoy, Cem

Published in:
Lecture Notes in Computer Science

DOI:
10.1007/11569596_7

2005

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

Link to publication

Citation for published version (APA):
Bür, K., & Ersoy, C. (2005). Admission control for multicast routing with quality of service in ad hoc networks. In P. Yolum, T. Güngör, F. Gürgen, & C. Özturan (Eds.), *Lecture Notes in Computer Science* (Vol. 3733, pp. 44-53). Springer. https://doi.org/10.1007/11569596_7

Total number of authors:
2

Admission Control for Multicast Routing with Quality of Service in Ad Hoc Networks

Kaan Bür and Cem Ersoy

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

Abstract. Ad hoc networks, being able to organize themselves without user intervention, can easily provide their users with mobility, multimedia support and group communication. However, they have to combine quality of service (QoS) and multicast routing strategies. This article defines the resource management and admission control components of the ad hoc QoS multicast (AQM) routing protocol, which achieves multicast efficiency along the network. When nodes wish to join a session, a request-reply-reserve process ensures that an appropriate QoS route is selected. Nodes are prevented from applying for membership if there is no QoS path for the session. To cope with the continuous nature of multimedia, AQM nodes check the availability of bandwidth in a virtual tunnel of nodes. Objection queries are issued prior to admission in order to avoid excessive resource usage by the nodes which cannot detect each other. New performance metrics are introduced to evaluate AQM's member and session satisfaction rates. Simulation results show that AQM improves multicast efficiency both for members and sessions.

1 Introduction

The evolution of wireless communication technologies has reached a point where it is easy to integrate them to handheld computing devices. Today, a new generation of portable computers is available, offering users more computational power than ever, in addition to mobility, multimedia support and group communication. However, these devices confront consumers with the heavy task of configuration. It becomes increasingly important that, once a mobile device is operational, it is able to configure itself with networking capabilities, asking its users only for their personal preferences and making the administrative work transparent. This requirement popularizes ad hoc networks, which are self-organizing communication groups formed by wireless mobile hosts. They make their administrative decisions in a distributed manner without any centralized control. They are free from the boundaries of any existing infrastructure. They are considered for many applications, including group-oriented computing such as disaster relief, community events and game playing.

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

In order to meet the mobile users' quality of service (QoS) expectations for such applications, ad hoc networks need to manage their scarce resources efficiently, which makes admission control a fundamental requirement. Multicast routing can improve wireless link efficiency by exploiting the inherent broadcast property of the wireless medium. The advantage of multicast routing is that packets are only multiplexed when it is necessary to reach two or more receivers on disjoint paths. Combining the features of ad hoc networks with the usefulness of multicast routing, a number of group-oriented applications can be realized.

The ad hoc QoS multicast (AQM) routing protocol is presented as a composite solution to the problem [1], which tracks QoS availability for each node based on current resource reservations. In this article, the join process of AQM is enhanced with: (a) virtual tunnels of bandwidth to avoid excessive resource allocation; (b) objection queries to control admission. Simulations show that AQM significantly improves multicast efficiency for members and sessions through QoS management.

The rest of this article is organized as follows. Previous research related to QoS systems and multicast protocols in ad hoc networks is reviewed in Section 2. After a short summary of AQM, the virtual tunnel approach to bandwidth availability and the objection query mechanism for admission control are introduced in Section 3. The performance of the proposed system is evaluated in Section 4. Final remarks and future work are presented in Section 5.

2 Quality of Service Systems and Multicast in Ad Hoc Networks

A QoS system consists of several components, including service differentiation, admission control, and resource allocation [2, 3]. Service differentiation schemes use QoS techniques such as priority assignment and fair scheduling. Priority-based mechanisms change the waiting times of the frames and assign smaller values to high-priority traffic. Fair scheduling algorithms partition resources among flows in proportion to a given weight and regulate the waiting times for fairness among traffic classes [2]. Measurement-based admission control schemes observe the network status, whereas calculation-based mechanisms evaluate it using defined performance metrics. Without admission control, the provision of QoS only by differentiating flows and coordinating channel access order is not effective for high traffic loads [3]. A contention-aware admission control protocol (CACP) introduces the concept of an extended contention area covering the carrier sensing range of a node [4]. Admission decisions are based on the available bandwidth information collected from the neighbours in the contention area.

Another important feature of a QoS system is congestion control. Congestion occurs when the data sent exceeds the network capacity and causes excessive delay and loss. It is avoided by predicting it and reducing the transmission rate accordingly. If congestion is local, it can be handled locally by routing around the congested node without reducing the data rate [5]. A multicast congestion control scheme for multi-layer data traffic is applied at the bottlenecks of the multicast tree using the queue states [6]. Some flow information is maintained at each node, and data layers are blocked and released to solve congestion and adjust the bandwidth rate.

Various protocols are proposed to maintain a multicast graph and perform routing in ad hoc networks. However, they do not address the QoS aspect of the subject, which becomes important as the demand for mobile multimedia increases.

Independent-tree ad hoc multicast routing (ITAMAR) provides heuristics to find a set of independent multicast trees, such that a tree is used until it fails and then replaced by one of its alternatives [7]. Maximally independent trees are computed by minimizing the number of common edges and nodes. Some overlapping is allowed since totally independent trees might be less efficient and contain more links. Thus, the correlation between the failure times of the trees is minimal, which leads to improved mean times between route discoveries.

Lantern-tree-based QoS multicast (LTM) is a bandwidth routing protocol which facilitates multipath routing [8]. A lantern is defined as one or more subpaths with a total bandwidth between a pair of two-hop neighbouring nodes, whereas a lantern path is a path with one or more lanterns between a source and a destination. A lantern tree serves as the multicast tree with its path replaced by the lantern-path. The scheme provides a single path if bandwidth is sufficient or a lantern-path if it is not.

Probabilistic predictive multicast algorithm (PPMA) tracks relative node movements and statistically estimates future relative positions to maximize the multicast tree lifetime by exploiting more stable links [9]. Thus, it tries to keep track of the network state evolution. It defines a probabilistic link cost as a function of energy, distance and node lifetime. The scheme tries to keep all the nodes alive as long as possible. It models the residual energy available for communication for each node, which is proportional to the probability of being chosen to a multicast tree.

3 Admission Control in the Ad Hoc QoS Multicasting Protocol

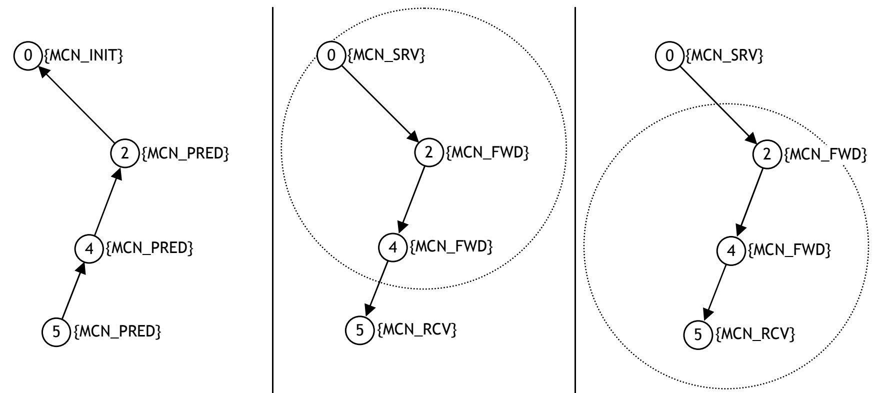
Since the main structure of the AQM protocol has been previously defined [1], the design details are not repeated below. Instead, following a short summary of AQM session management, special emphasis is laid on admission control and the means of dealing with mobility. The virtual tunnel approach to checking bandwidth availability and the objection query mechanism are introduced as enhancements to the protocol.

3.1 Session Management

When a node broadcasts a join request (JOIN_REQ) for a session, its predecessors (MCN_PRED) propagate the packet upstream as long as QoS 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. Tables of active sessions, known members and neighbours are also maintained at each node. A forwarded request eventually reaches members of that session which issue replies (JOIN_REP) back to the requester if QoS can be satisfied. Prior to replying, however, they send a one-hop objection query (JOIN_OBJ) to their neighbours to check if a possible new resource allocation violates the bandwidth limitations of these. The objection query mechanism is explained in Section 3.3. Downstream nodes that have forwarded join requests forward the replies towards the requester. During this process, they also exploit the objection query mechanism since it is possible that they qualify as forwarders. The originator of the join request selects the one with the best QoS conditions among the replies it receives. It changes its status from predecessor to receiver (MCN_RCV) and sends a reserve message (JOIN_RES) to the selected node. The reserve packet propagates along the selected path and finally reaches the originator of the reply. Intermediate nodes on the path become forwarders (MCN_FWD). If this is the first receiver, the session initiator (MCN_INIT) becomes an active server (MCN_SRV).

3.2 The Virtual Tunnel of Bandwidth

The continuous nature of multimedia applications requires a new method of checking bandwidth availability to see if the QoS requirements of a new join request can be met. Being within the transmission range of each other, a session server about to allocate resources for its first member and the forwarding node immediately following it share the bandwidth of the same neighbourhood. Therefore, a server has to ensure that its successor also has enough bandwidth available to forward multicast data packets that it receives. In other words, twice as much bandwidth has to be available in the neighbourhood than the amount required by the QoS class of the session.



(a) Join request of node n_5 | (b) View of replying node n_2 | (c) View of replying node n_4

Fig. 1. The virtual tunnel approach to checking bandwidth availability: (a) JOIN_REQ of n_5 propagates towards n_0. Prior to sending a JOIN_REP, n_0 checks for two times QoS bandwidth since it has to ensure that n_2 can also forward packets. (b) n_2 checks for three times QoS bandwidth since, in addition to its predecessor n_0 and itself, it has to ensure that n_4 can also forward the data. (c) Finally, n_4 checks for two times QoS bandwidth since n_5 is only a receiver.

Following the path downstream towards the new member, 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 successor. A similar judgement can be made for the rest of the intermediate nodes. Fig. 1 shows the virtual tunnel approach to checking the bandwidth availability.

Thus, nodes have to check for availability of the necessary bandwidth according to their position within the multicast tree before accepting a new request. When it is time to reserve resources, however, each node is responsible only for itself, i.e., nodes allocate only the amount of bandwidth that is necessary for the session of a particular QoS class. For a member already forwarding packets of that session, this requirement is met automatically since the node has already been through this allocation process.

3.3 The Objection Query Mechanism

A node decides whether or not to take part in a session as a forwarder based on its current resource availability. While this approach prevents the node from overloading itself, it is not enough to help other nodes balance their loads. Although a node does not allocate more bandwidth than available in its neighbourhood, the overload problem arises as a result of the admissions made by its neighbours which cannot directly detect each other. In other words, a node can be surrounded by several neighbours, some of which are not within the transmission range of each other. The node experiences overload due to excessive resource usage in its neighbourhood, which cannot be foreseen since the surrounding nodes are not aware of each other's reservations. To overcome this problem, each replying node consults its neighbours first to see if any of them becomes overloaded. This is necessary since it is otherwise impossible for a node to see the bandwidth usage beyond its direct neighbours.

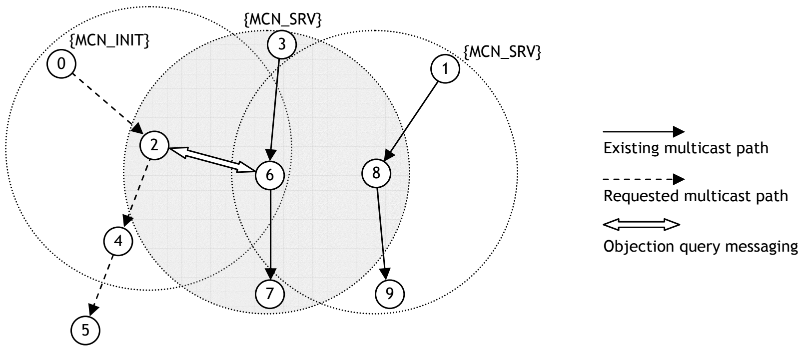


Fig. 2. The objection query mechanism: During the reply phase of a join process, the nodes n_0, n_2 and n_4 issue one-hop objection queries before sending their replies. At the time n_2 sends its query, n_6 is already sharing the bandwidth in its neighbourhood with n_3, n_7 and n_8. However, n_2 is not aware of this since it cannot directly detect the others. Thus, n_6 objects to n_2 offering their common resources to n_4 if the total allocation exceeds the capacity of n_6's neighbourhood.

A node having received a reply issues an objection query prior to forwarding the reply. This one-hop message containing information on the requested bandwidth allows the neighbours to object to a possible data flow along this path, if they start suffering from overload as a result of the allocation. If the new reservation causes the limit to be exceeded, the neighbour sends the objection to the node which has queried it. Otherwise the query is discarded. If the node having sent the query receives any objection, it discards the reply. Otherwise the query times out, indicating that the new node can be safely admitted. Only those neighbours who are serving one or more sessions may object to new allocations. It is not important that a silent node becomes overloaded. Fig. 2 shows a situation where the objection query mechanism is utilized.

A session initiator, which is about to get its first member, or an intermediate node about to forward a reply towards a requester have to issue an objection query first. An active member forwarding packets of a session does not need to query objections for each new join request since it has previously consulted its neighbours.

3.4. Dealing with Mobility

One of the major concerns for ad hoc communications is the ability of the routing infrastructure to cope with the dynamics of node mobility. In order to maintain connectivity and support QoS with maximum possible accuracy under mobility conditions within their neighbourhood, nodes perform periodic update and cleanup operations on their session, membership and neighbourhood tables.

The session information is refreshed periodically via session update packets (SES_UPDATE) sent by the session initiator. They are propagated once as long as the QoS requirements of the session can be fulfilled, even if they belong to a previously known session and come from a known predecessor to ensure that all new nodes in a neighbourhood are informed on the existence of the ongoing sessions they can join.

Lost neighbours are removed from the neighbourhood, session, membership and request tables. 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 forwarder or a receiver, loses its preceding forwarder or server, this means that it loses its connection to the session. It changes its own status to a predecessor, i.e., a regular node which is aware but not an active member of the session. It also informs its successors with a lost session message (SES_LOST) if it is a forwarding member of the session. Downstream nodes receiving the lost session messages interpret them similarly to update their status regarding the lost session and forward the message if necessary. This mechanism, combined with the periodic updates, keeps nodes up-to-date regarding the QoS status of the sessions and ready for future membership admission activities. It also prevents them from making infeasible join attempts.

4 Computational Performance Experiments

The simulations are conducted using OPNET Modeler 10.5 Educational Version with the Wireless Module [10]. They are repeated 20 times for each data point and results are aggregated with a 95% confidence interval for a multicast scenario with four QoS classes representing a sample set of applications. Nodes initiate or join sessions according to a certain probability. Generated sessions are assigned randomly to one of the four QoS classes defined in Table 1. Thus, the ad hoc network supports four types of multicast applications simultaneously and manages the QoS requirements of each application depending on its class definition. To comply with the sample bandwidth and delay bounds given as part of these QoS class definitions, nodes are restricted to certain minimum bandwidth and maximum hop count regulations. In other words, a node may join a session only if it can find a path to the server with more bandwidth available than the minimum and less hops away than the maximum allowed.

A node can take part at only one application at a time as a server or receiver, whereas it can participate in any number of sessions as a forwarder as long as QoS conditions allow. Apart from that, there is no limit to the size of the multicast groups. The effect of mobility on the performance of AQM is observed under the random waypoint mobility model. In contrast to previous performance evaluations, which limit their simulations to a few minutes and a single session, four hours of network lifetime have been simulated to get a realistic impression of the behaviour of multiple multicast sessions being maintained simultaneously in a distributed manner. The parameters of the mobility model and other simulation settings are given in Table 2.

Table 1. QoS classes and requirements

QoS Class	Bandwidth Requirement	Average Duration	Delay Tolerance	Relative Frequency	Application Type
0	128 Kbps	1,200 s	Low	0.4	High-quality voice
1	256 Kbps	2,400 s	High	0.2	CD-quality audio
2	2 Mbps	1,200 s	Low	0.3	Video conference
3	3 Mbps	4,800 s	High	0.1	High-quality video

Table 2. Simulation parameters

Parameter Description	Value
Area size	1,000 m x 1,000 m
Greeting message interval	10 s
Maximum available link bandwidth	10 Mbps
Mobility model	Random waypoint
Node speed	1-4 m/s (uniform)
Node pause time	100-400 s (uniform)
Node idle time between sessions	300 s (exponential)
Session generation / joining ratio	1 / 9
Session update message interval	60 s
Wireless transmission range	250 m

The evaluation of QoS multicast routing performance in ad hoc networks requires novel criteria that are both qualitative and measurable. The main concern of this article is to test the efficiency of AQM in providing multicast users with QoS and satisfying the service requirements of multimedia applications. Therefore, it is necessary to focus on member satisfaction. The member overload avoidance ratio O_{Member} is introduced as a new performance metric in terms of QoS requirements, which is the number of overloaded nodes o divided by a weighted sum of the number of servers s and forwarders f, subtracted from the maximum possible unit ratio of 1:

$$O_{Member} = 1 - \frac{o}{s + \alpha f} \ . \tag{1}$$

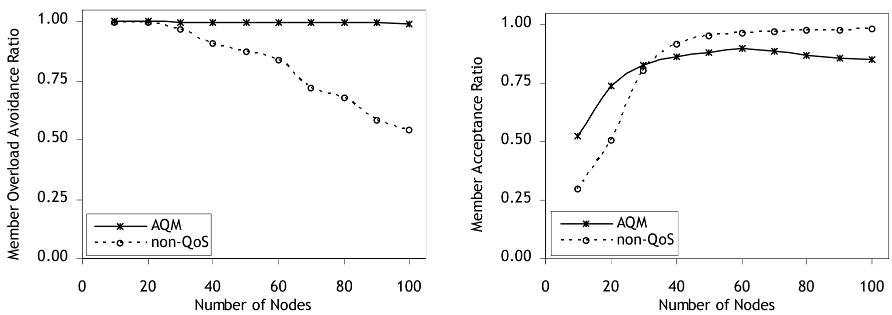
The coefficient α points out that the impact of overloaded nodes on forwarders is greater than that on servers, due to the fact that the former are intermediate nodes affected by both their predecessors as well as their successors.

The efficiency of ad hoc multicast routing protocols is typically measured by the session success rate, or the member acceptance ratio A_{Member}, which is defined as the number of accepted receivers r divided by the number of session join requests q:

$$A_{Member} = \frac{r}{q} \ . \tag{2}$$

It should be noted that O_{Member} and A_{Member} present a trade-off with regard to member satisfaction. While improving the former with QoS restrictions, an efficient QoS multicast routing protocol should be able to keep the latter at an acceptable level.

Fig. 3(a) compares the member overload avoidance ratio of AQM to the non-QoS scheme, where $\alpha = 0.5$. In AQM, where QoS support is active, nodes do not make allocations exceeding the maximum bandwidth available in their neighbourhood. The number of overloaded members is kept to a minimum with the introduction of the objection query mechanism. In the non-QoS scheme, nodes accept join requests if they can find a path towards the session server. Since they do not care about available resources, they soon become overloaded. As the number of network nodes grows, more sessions are initiated, and more requests are accepted without considering the available bandwidth, which causes a drastic decrease in the ratio of members not overloaded for the non-QoS network. The results show that AQM outperforms the non-QoS scheme with its ability to prevent members from being overloaded.



(a) Overload avoidance (b) Member acceptance

Fig. 3. AQM vs. the non-QoS scheme with regard to member satisfaction.

Fig. 3(b) compares the member acceptance ratio of AQM to the non-QoS scheme. A decrease in the member acceptance of AQM is expected as a result of the tight resource management and admission control precautions taken by the protocol. However, in networks with a small number of nodes and low connectivity, AQM performs even better than the non-QoS scheme since it informs its nodes periodically on the availability of ongoing sessions and prevents them from making requests for sessions that are not reachable any more. As the network density grows and more requests are made, the performance of AQM remains close to the non-QoS scheme. In

AQM, where QoS restrictions apply, nodes do not accept new requests if they cannot afford the required free bandwidth. Thus, not all requests are granted an acceptance and the member acceptance ratio is lower than the non-QoS scheme. However, AQM is still able to achieve an acceptance ratio close to the non-QoS scheme due to its ability to eliminate infeasible join requests before they are issued by keeping its nodes up-to-date regarding the QoS conditions in the network and the status of the sessions.

It is inevitable that the computational overhead of a routing protocol increases with its complexity. However, it is possible to keep it at an acceptable level while adding QoS functionality to the protocol. The member control overhead of a multicast session member C_{Member} is formulated as the number of control packets processed p divided by sum of s, f and r, which gives the number of active nodes in the network, participating in at least one multicast session as a server, a forwarder, or a receiver:

$$C_{Member} = \frac{p}{s + f + r} \ . \tag{3}$$

Fig. 4 compares the member control overhead of AQM to the non-QoS scheme. In addition to the periodic session update packets, AQM uses lost session notifications to keep nodes up-to-date with regard to session availability. It sends one-hop objection queries to ensure that session members do not become overloaded. As the network population grows, more of these packets are necessary since more multicast paths are possible in a more crowded network. It can be concluded from the figure that AQM provides QoS with an acceptable overhead. In fact, by rejecting some of the join requests, AQM cuts further communication with those nodes, whereas the non-QoS scheme communicates with all requesters until their routing information is delivered.

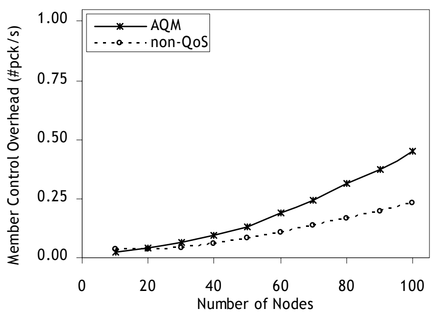


Fig. 4. AQM vs. the non-QoS scheme with regard to member control overhead.

5 Conclusion

The increasing amount of multimedia content shared over wireless communication media makes QoS-related, resource-efficient routing strategies very important for ad hoc networks. AQM provides ad hoc networks with these features. It keeps the network up-to-date on the availability of sessions with regard to QoS considerations.

It controls the availability of resources throughout the network and ensures that users do not suffer from QoS degradation. AQM takes the continuity property of multimedia data into consideration and checks bandwidth availability along a virtual tunnel of nodes. It also facilitates an objection query mechanism to inform nodes on possible overload on others. AQM also sets limits to path length in terms of hop count and checks them in order to satisfy the delay requirements. Thus, it utilizes efficient admission control mechanisms, sustains QoS along the ad hoc network and eliminates infeasible membership requests proactively at their sources.

Service satisfaction is the primary evaluation criterion for a QoS-related scheme. Simulations give a good insight to the quality of AQM. By applying QoS restrictions, AQM achieves lower overload on members and improves the multicast efficiency for members and sessions. Without a QoS scheme, users experience difficulties in getting the service they demand as the network population grows and bandwidth requirements increase.

A future research direction for this work is the assessment of the recent multicast routing protocols to have an alternate view to their performance in terms of QoS as experienced by the user. A second topic is the efficient rerouting of multicast sessions when changes occur in the network topology as a result of mobility or varying QoS conditions. It is also a good idea to evaluate ad hoc network protocols with multiple mobility models. Ad hoc applications with team collaboration and real-time multimedia support necessitate group mobility, which improves performance if protocols take advantage of its features such as multicast routing.

References

1. Bür, K., Ersoy, C.: Ad Hoc Quality of Service Multicast Routing. Elsevier Science Computer Communications, to appear, (2005)
2. Pattara-Atikom, W., Krishnamurthy, P.: Distributed Mechanisms for Quality of Service in Wireless LANs. IEEE Wireless Communications **10**(3) (2003) 26-33
3. Zhu, H., Li, M., Chlamtac, I., Prabhakaran, B.: A Survey of Quality of Service in IEEE 802.11 Networks. IEEE Wireless Communications **11**(4) (2004) 6-14
4. Yang, Y., Kravets, R.: Contention-Aware Admission Control for Ad Hoc Networks. IEEE Transactions on Mobile Computing **4**(4) (2005) 363-377
5. Barry, M., Hogan, B.J., McGrath, S.: Congestion Avoidance in Source Routed Ad Hoc Networks. In: Proceedings of the IST Mobile Summit, Lyon, France, (2004) 682-686
6. Peng, J., Sikdar, B.: A Multicast Congestion Control Scheme for Mobile Ad-Hoc Networks. In: Proceedings of IEEE Globecom, San Francisco, USA, (2003) 2860-2864
7. Sajama, S., Haas, Z.J.: Independent-Tree Ad Hoc Multicast Routing (ITAMAR). ACM Mobile Networks and Applications **8**(5) (2003) 551-566
8. Chen, Y.S., Ko, Y.W.: A Lantern-Tree-Based QoS On-Demand Multicast Protocol for a Wireless Mobile Ad Hoc Network. IEICE Transactions on Communications **E87-B**(3) (2004) 717-726
9. Pompili, D., Vittucci, M.: A Probabilistic Predictive Multicast Algorithm in Ad Hoc Networks (PPMA). In: Proceedings of MED-HOC-NET, Mugla, Turkey, (2004)
10. OPNET Technologies Inc, Bethesda, MD, USA. available at http://www.opnet.com