Ad Hoc Networks 9 (2011) 48–60

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

MARS: Link-layer rate selection for multicast transmissions in wireless mesh networks

Prashanth A.K. Acharya *, Elizabeth M. Belding

Department of Computer Science, University of California, Santa Barbara, CA 93106, United States

ARTICLE INFO

Article history:
Received 3 August 2009
Received in revised form 6 January 2010
Accepted 4 May 2010
Available online 10 May 2010

Keywords:
Wireless mesh networks
Rate selection
IEEE 802.11 multicast

ABSTRACT

IEEE 802.11 devices dynamically choose among different modulation schemes and bit-rates for frame transmissions. This rate adaptation, however, is restricted only to unicast frames. Multicast (and broadcast) frames are constrained to use a fixed low bit-rate modulation, resulting in low throughput for multicast streams. Availability of high bandwidth and efficient use of the medium is crucial to support multimedia multicast streaming applications such as IPTV, especially in multihop mesh networks. To address this problem, we propose a rate adaptation algorithm for multicast transmissions in these networks. The proposed algorithm, MARS, is distributed in nature, and relies on local network measurements to select a transmission bit-rate for a given multicast group. The algorithm also facilitates the joint use of bit-rate selection and link-layer mechanisms such as acknowledgements and retransmissions to improve reliability of high throughput multicast streams. Based on implementation and evaluation on a testbed, the algorithm provides up to 600% gain in throughput compared to traditional 802.11 networks in some scenarios. Additionally, the algorithm can support multicast streams while consuming a small fraction (20%) of the resources compared to the basic 802.11 operation.

1. Introduction

Wireless mesh networks (WMNs) are touted to provide the next generation of ubiquitous network connectivity. IEEE 802.11-based WMNs enable service providers to extend connectivity to a wide geographical region with minimal cost by avoiding installation of cables and expensive base-stations. Numerous towns and cities around the world have deployed or plan to deploy IEEE 802.11-based WMNs to blanket the region with WiFi access. A parallel trend in network usage is the increased popularity of multimedia applications such as voice and video. The explosive growth in the usage of websites such as youtube.com is a testament to this trend. As the usage of these applications increases, it becomes necessary to support these applications on WMNs as well.

A key characteristic of several multimedia streaming applications is their use of multicast transmissions for transportation of content. Multicast streams are particularly important when several clients wish to receive the same audio/video stream, e.g., a live stream of a football match, IP-based radio station, IPTV, webcast of a live TV (SkyPlayer from Sky Networks, UK). The ITU focus group for IPTV standardization recommends the use of IP multicast for the live video mode of operation of IPTV [1]. Multicast streams are also supported by popular multimedia streaming servers such as Windows Media Services, SHOUTCast and VLC. These multicast streams are characterized by their requirement of high bandwidth and low latency. In addition, these streams cannot tolerate high packet loss rates. In this context, we examine the ability of 802.11 WMNs to support high bandwidth, low latency, low loss multicast streams.

* Corresponding author.
E-mail addresses: acharya@cs.ucsb.edu (P.A.K. Acharya), ebelding@cs.ucsb.edu (E.M. Belding).

1570-8705/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.adhoc.2010.05.001

In a multi-rate 802.11 network, devices are capable of transmitting packets using different modulation schemes and bit-rates varying from 1 Mbps to 54 Mbps. These devices dynamically select an appropriate transmission bit-rate using a *rate adaptation* algorithm to obtain high bandwidth. However, as per the 802.11 standard, the use of rate adaptation algorithms is currently restricted to unicast frames only. Multicast and broadcast transmissions are forced to use a fixed lower modulation bit-rate, usually the lowest bit-rate of 1 Mbps in a 802.11b/g network.[1] This restriction has two effects related to multicast streams. First, it inherently limits the capacity available for a multicast stream, and thereby the maximum achievable bandwidth. Second, a stream that uses a bit-rate of 1 Mbps consumes a disproportionate fraction of *airtime* for transmission of its packets, and therefore may adversely impact other existing flows in the network, possibly even causing network congestion. Further, 802.11 multicast frames, unlike unicast frames, are transmitted without requiring an acknowledgement (ACK frame) from the receiver(s). This absence of feedback from the receiver inhibits the automatic error recovery through link-level retransmissions, and directly impacts the reliability of the multicast application streams. For these reasons, multicast-based streaming applications have generally been considered unsuitable for 802.11-based WMNs. Therefore, in order to support high throughput multicast streams, there is a need for a method to transmit multicast frames with low cost to the network. In addition, it is desirable to increase the degree of link-level reliability for multicast packets, similar to that provided by retransmissions for unicast frames.

To this end, we propose Multicast Auto Rate Selection (MARS), wherein multicast transmissions in a wireless mesh network utilize higher modulation rates in order to increase the throughput of multicast flows. Our motivation for this approach is driven by the insight that mesh routers in most networks are deployed with careful planning, and in such networks, links among neighboring mesh routers are frequently of high quality to support higher modulation rates. Further, the mesh backhaul nodes are stationary, and link quality among neighbors can be measured quite accurately. MARS is a measurement driven scheme that actively tracks the quality of the links between a mesh router and its neighbors. Using this measurement, MARS selects the best transmission rate for each multicast group based on link quality to the members of that group. We equip MARS with the ability to retransmit multicast frames to increase link-level redundancy and thereby improve reliability. This flavor of the protocol is called MARS-R. The transmitter node selects one of the receiver nodes to respond with an ACK on successful packet reception. These schemes address the challenge of maximizing end-end throughput using a distributed solution, given the diverse set of link qualities throughout the network.

Our contributions in this work are as follows:

- We motivate the case for rate adaptation of multicast frame transmissions in 802.11-based WMNs, via an analysis of link qualities in real networks.
- We propose MARS, a distributed measurement-driven rate adaptation approach for multicast transmissions. Further, the MARS-R enhancement enables retransmissions for such frames.
- We provide a practical implementation of MARS. Performance evaluation on a testbed shows that we are able to obtain about 600% increase in maximum end-end throughput for some scenarios. MARS provides the same throughput performance as traditional 802.11 but requires only 20% of the transmission time.
- We find that the use of retransmissions and ACKs in MARS-R significantly increases the packet delivery ratio, and the associated additional overhead does not adversely affect performance.

The remainder of the paper is organized as follows. Section 2 surveys related work. We motivate the case for multicast rate adaptation in Section 3. The details of the design and implementation of MARS are presented in Sections 4 and 5, respectively. We evaluate the performance of the algorithm in Section 6. Section 7 concludes the paper.

We note that throughout the paper the term *bit-rate* refers to the physical modulation rate used to encode the 802.11 frames, *frame* refers to a MAC layer entity, and *packet* refers to a network/application-layer entity.

2. Related work

A large body of research has studied multicast flow support in multihop networks. Several protocols have been proposed to address the problem of routing multicast flows in a multihop network [2–5]. These solutions are designed for MANETs, and hence are mainly focused on connectivity. A recent work by Roy et al. focuses on multicast routing with the goal of forming high throughput routes in stationary multihop networks [6]. The authors propose the use of the Multicast Expected Transmission Count (METX) metric for this purpose. However, they assume fixed bit-rate transmissions for multicast frames. The throughput improvement achieved by the METX metric is therefore constrained by the multicast capacity at the MAC layer. Other solutions have suggested adapting the packet generation rate at the application-layer to react to changes in network conditions [7]. Our focus is on the link-layer behavior of multicast transmissions, and therefore complementary to the above approaches.

Several researchers have proposed solutions to solve the *broadcast storm* [8] problem in multihop wireless networks by reducing the number of transmissions required for network-wide flood of a message [9–11]. Wieselthier et al. highlight the importance of exploiting the *wireless multicast advantage* in the context of energy efficient multicasting [12]. They suggest utilizing the broadcast nature of wireless transmissions instead of using link-abstractions in order to minimize the number of packet transmissions. Gandhi et al. prove that the problem of finding a minimum latency broadcast tree in a multihop network is NP-Hard

[1] Although there is no documented reason motivating this restriction, we believe this was done to ensure backward-compatibility with legacy 802.11 or 802.11b-only devices. Another reason could be to maximize the communication range and reliability of broadcast transmissions.

[13]. They suggest the use of approximation algorithms to determine low latency distribution trees.

In the context of multi-rate mesh networks, Chou et al. have suggested the use of higher bit-rates for broadcast frame transmissions in order to minimize broadcast latency [14,15]. They show that the joint problem of creation of a broadcast distribution tree and selection of transmission bit-rate is also NP-Hard, and present heuristic-based solutions for the problem. The authors propose the use of a Rate-Area Product metric that captures the tradeoff between the bit-rate used and the number of neighbors reached. Their approach of minimizing delay is complementary to our methodology of increasing throughput. However, we note that their algorithm is centralized, in contrast to our distributed algorithm. Further, previous research has shown that the range-reliability relationships do not always hold true in real environments. Our algorithm deals with links that exhibit reliability of less than 100%.

The problem of rate adaptation for unicast transmissions in 802.11 networks has also been well studied and several algorithms have been proposed [16–19]. These algorithms rely on the ACK frame for 802.11 unicast frames to learn about the packet delivery success. However, multicast frame transmissions are characterized by the absence of both an ACK frame and retransmissions. These algorithms, therefore, cannot be applied directly to the problem of multicast rate adaptation. However, we utilize the concept of probing and collecting statistics of the various bit-rates, proposed in SampleRate [18], as described in Section 4.

Kuri and Kasera propose the use of a leader-based protocol to enable the use of control messages such as RTS, CTS, ACK and NACK for multicast transmissions in a WLAN [20]. One of the receivers is elected to send feedback to the transmitter. Their approach, however, requires an expensive RTS–CTS exchange prior to each packet. Further, the transmission of a new type of control packet (NACK) may not be possible on commodity hardware. Villalón et al. build upon the leader-based scheme to incorporate multiple bit-rates for WLAN multicast transmissions [21]. This solution also uses the NACK packet. Their choice of bit-rate is based on signal strength, a metric that is hard to measure reliably in a multihop network [19]. In the context of network coding, Katti et al. propose the use of pseudo-broadcast frames (i.e., broadcast payload wrapped in unicast frames) to enable link-layer ARQ for frames with multiple receivers [22].

3. Motivation

In this section we study the quality of links as seen in two mesh network deployments – the MIT Roofnet [23] and UCSB MeshNet [24]. We argue the case for the feasibility of the use of higher bit-rate transmissions for multicast frames in a 802.11 mesh network. The basic methodology is as follows. We know from previous research, that for a given stationary link, the probability of a packet loss increases with higher modulation rates (i.e., higher bit-rates) [17]. We study the packet loss rates (or inversely, the packet delivery rates) at different bit-rates in these real world networks. We show that frequently the packet delivery rate (PDR) at higher bit-rates is comparable to that at the default multicast bit-rate of 1 Mbps for links used in unicast routes. In these cases, a higher bit-rate can be used to reduce delivery delay while maintaining the same (or even greater) throughput.

3.1. MIT Roofnet

The MIT Roofnet is an unplanned community-based 802.11 mesh network in Cambridge, MA [23]. The network consists of 38 randomly deployed nodes. We analyze the link-layer reliability information for the month of August 2004 from the Roofnet dataset [23]. In particular, we study the packet delivery ratio for all the links at different link bit-rates. The PDR for each link was recorded with the help of transmission of 1500 byte packets at the four different 802.11b bit-rates for 90 s each. To compute the PDR, each Roofnet node broadcasts a 1500 byte probe at each of the four 802.11b data rates, and a 134 byte probe at 1 Mbps. The 1500 byte probe is used to estimate the delivery probability of a large data packet, whereas the 134 byte probe is used for the 802.11 ACK frame [25].

To understand the utility of using higher bit-rates for multicast transmissions, we only consider the forward PDR; we do not consider the PDR in the reverse direction (for the ACK frame in the unicast case) for our analysis. We intuit that a link can use a higher bit-rate (say 11 Mbps) than the default bit-rate (1 Mbps) if the PDR of the higher bit-rate is similar to the PDR at 1 Mbps. Most multicast applications can be designed to tolerate some link-layer packet loss. In other words, a higher bit-rate can be used if it does not lead to a significant number of packet losses. Therefore, for each link, we define r_i to be the highest bit-rate such that the forward PDR at r_i is at least ϕ. We set $\phi = 85\%$ to model an example sustainable packet loss rate for multimedia applications that use the H.263 codec with loss-resilient encoding [26]. If none of the bit-rates have PDR more than ϕ, the r_i of the link is 1 Mbps, the default multicast bit-rate. Since Roofnet is an 802.11b network, r_i can either be 1, 2, 5.5 or 11 Mbps. We count the number of links in the network at each r_i, and plot the histogram of this data in Fig. 1a for all links. The results show a large fraction (>85%) of the links have an r_i of 1 Mbps, indicating high packet loss rates.

For the next part of the analysis, we consider the subset of links used for unicast routes to the gateway. These links form an instance of a multicast distribution tree that covers all the nodes in the mesh network, and is rooted at the gateway node. In order to determine these unicast routing links, we simulate the behavior of the routing protocol *Srcr* used in Roofnet [27]. The routing metric used in Srcr is Expected Transmission Time (ETT). ETT is calculated using the bi-directional PDR at each bit-rate. Each node chooses the gateway with the best throughput (i.e. lowest ETT metric). Based on these route calculations, we obtain the subset of links used for unicast routing from every node in the network to its chosen gateway. We then perform the above described highest bit-rate r_i distribution analysis for this subset of links. This data is also plotted

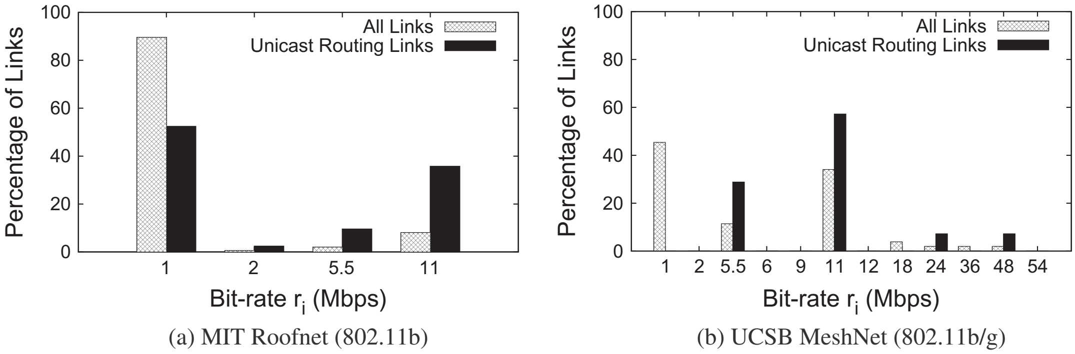

Fig. 1. Histograms of multicast bit-rate: distribution of the number of links at each bit-rate r_i, such that r_i is the highest bit-rate with PDR at least 85%.

in Fig. 1a. When we consider only the unicast route links, we observe that r_i for 50% of the links is higher than 1 Mbps. In particular, about 40% of the links have r_i of the highest bit-rate 11 Mbps, i.e., the PDR at 11 Mbps on these links is more than ϕ. This observation implies that a multicast distribution tree that is comprised of links from the unicast routes can use high bit-rates without any significant impact on the PDR. The reason for this phenomenon is that the throughput-based routing protocol favors high throughput links for use in routing traffic. These links are characterized by high PDR even at higher bit-rates. A similar analysis with ϕ = 80% and ϕ = 90% showed that more than 60% and 35% of unicast routing links had r_i of more than 1 Mbps, respectively.

3.2. UCSB MeshNet

The UCSB MeshNet is a 15-node indoor mesh network testbed with nodes deployed on three floors of a building [24]. Each node is equipped with two Atheros chipset-based IEEE 802.1a/b/g radios, and the radios are controlled with the MadWifi driver (v0.9.3.3) for Linux (kernel 2.6). The primary radio on each node operates in the 802.11b/g mode and is dedicated to forming the multihop mesh backhaul connecting to the gateway node. The second radio is tuned to an orthogonal 802.11b/g channel and is used as a client access layer. The MIT Roofnet software is used for routing on the mesh backhaul layer.

In order to collect link reliability data, each node periodically sends broadcast probes of 512 bytes at each bit-rate. The node cycles through the 12 bit-rates of 802.11b/g in increasing order in a period of 12 s (i.e., at one probe per second). Five trials of 10 minutes durations were conducted. Each node also records the beacons that it received at each bit-rate for every neighbor. At the end of the experiment, we analyze the data in a manner similar to the Roofnet analysis. We calculate the highest bit-rate r_i such that the PDR at r_i is at least ϕ = 85% . This bit-rate r_i is computed for all the links in the network including links used in unicast routing paths (as shown by the routing software logs). The results are averaged over the five trials of the experiment. The histogram of link r_i bit-rates is plotted in Fig. 1b. When we consider all the links in the network, we observe that about 45% have an r_i (highest multicast bit-rate) of 1 Mbps. In the second part of the analysis, where we consider only the links used in unicast routing, we note that *all* the links have an r_i of more than 1 Mbps. In particular, about 57% of the links have an r_i of 11 Mbps. These values of r_i reflect the high quality of links available for unicast routing in the mesh.

The previous observations from the UCSB MeshNet and the MIT Roofnet motivate the feasibility of using higher bit-rates for multicast transmissions, especially when the multicast distribution tree is based on the unicast routing links. We use the insights obtained from this analysis in the design of our rate adaptation scheme described in the next section.

4. Design of MARS

At a high level, Multicast Auto Rate Selection (MARS) is an algorithm that chooses the best transmission bit-rate for a given multicast group. The algorithm is distributed in nature, and at each node in the mesh network, link quality information obtained from the immediate neighborhood is used to decide the transmission bit-rate. In the following sections, we first outline the assumptions in the design of MARS. Then, we present a detailed description of the design.

4.1. Assumptions

We consider the environment of a multi-layered 802.11 mesh network with a logical/physical separation of the mesh backhaul layer and the client access layer: an architecture used by many commercial vendors[2] as well as research networks (MIT Roofnet [23]). The problem under consideration is the transmission of a stream of link-layer multicast frames in the mesh backhaul layer. Higher layer protocols such as IGMP are assumed to handle group management functions such as joining or leaving a multicast group. Further, a multicast routing protocol sets up the distribution trees, i.e., designates the nodes that receive and forward packets. We assume that MARS has access to this information, i.e., for every active multicast group, each mesh

[2] Strix Systems – http://www.strixsystems.com, BelAir Networks - http://www.belairnetworks.com, FireTide – http://www.firetide.com.

node knows whether it is a forwarder, a receiver, or neither. In the case of a multicast forwarder, MARS has information regarding the downstream neighbors in the distribution tree. Similarly, a forwarder/receiver node knows its upstream neighbor in the distribution tree.

We assume that the root of the multicast forwarding tree is the gateway node of the mesh. This models the scenario wherein the source of the multimedia stream is an Internet host, and clients in the mesh network are subscribed to this stream. If the source of the stream is inside the mesh, we assume the gateway node is the rendezvous point for the multicast distribution tree. In accordance with the analysis presented in Section 3, we assume that the distribution tree rooted at the gateway node is formed by unicast routes from the gateway to each mesh node. Such a tree implies that for every mesh node, there is only one upstream neighbor. The current operation of MARS is designed on this assumption. A simple extension can enable MARS to operate in the presence of multiple upstream neighbors for different multicast streams. In the future, we intend to explore the problem of joint optimization of multicast tree formation and multicast rate adaptation, with throughput maximization as the criteria. As explained in [14], this is an NP-Hard problem.

4.2. Design overview

We consider the scenario wherein the mesh routers are part of a IP multicast group. Within the scope of the mesh network, the packets from the group are first transmitted by the gateway node. The packets are then forwarded by the mesh routers, so that every router that is part of the group receives this packet. At each hop in the mesh network, the packet is transmitted as a multicast 802.11 frame. A multicast 802.11 frame is different from a unicast 802.11 frame in two fundamental ways: (1) There is no feedback regarding the successful reception of the packet in the form of ACK frame; and (2) multicast frames are not retransmitted. The frame is destined to the multicast MAC address corresponding to the group IP address [28].

The default behavior of 802.11 is to choose a fixed bit-rate for transmission of every multicast frame. This bit-rate is generally the lowest common bit-rate (1 Mbps in 802.11b/g network and 6 Mbps in 802.11a). Rather than use a fixed low rate, MARS enables each mesh node to dynamically select the bit-rate for multicast transmissions. Since each multicast distribution tree may have different member nodes, the set of downstream neighbors for a given upstream node may be different for each group. MARS selects the best bit-rate for each multicast group separately depending on the downstream neighbors. The bit-rate selection is recomputed periodically to account for interference and environmental variations, as well as in response to changes in the multicast distribution tree. The algorithm is distributed, i.e., the bit-rate is computed at each mesh node based on information from the local neighborhood.

In order to select the best bit-rate for transmission of the multicast frame, the metric used by MARS is based on statistical data consisting of the packet delivery ratio (PDR) at the various bit-rates to each of its neighbors. The lack of feedback from the downstream neighbors, i.e., no ACK frames, means that upstream forwarding nodes do not have local information regarding the success of a frame transmission. Therefore, upstream nodes require explicit feedback information from the downstream nodes to learn about the packet delivery ratio at each bit-rate. Accordingly, upstream nodes send periodic probes at each bit-rate. Downstream nodes record these probes, calculate PDRs, and send this information to the upstream nodes. Upstream nodes use this periodic feedback to determine the multicast bit-rate for each multicast group. The main goal of the bit-rate selection at an upstream node is to maximize end-end throughput given the different combinations of PDR and bit-rate.

The functionality of MARS consists of two main components. The first component involves the collection of packet delivery rates (PDRs) at the bit-rates for the downstream links in the multicast tree. The second component consists of using this data to choose a bit-rate for transmission. We now describe each of these components in detail.

4.3. Probing and data collection

Each mesh node sends periodic probe packets as a 1-hop MAC layer broadcast at different bit-rates in order to measure the PDR at each bit-rate. The probe traffic is not specific to an individual multicast group. Instead, the local broadcast probe ensures that an upstream node has information regarding all it neighbors. The node can then select the appropriate bit-rate for each multicast group using the common probe data. The probes are sent at the rate of N probes per second (total over all the bit-rates). We study the impact of this parameter in Section 6.4.

Each probe packet contains two types of information. The first type is information regarding the sender, and includes the sender's unicast IP address, timestamp, probe sequence number, and the transmission bit-rate. The second part of the probe packet includes reception information of previous probe packets sent by neighbors of the sender. The packet reception information is essentially a record of the number of packets received at each bit-rate in the last period. The packet reception record for a neighbor is of the following format: <NeighborAddress, p_1,p_2,$p_{5.5}$,…,p_{54}, LastSequenceNumber>. p_i indicates the number of probes received at bit-rate i in the last λ seconds. LastSequenceNumber is the sequence number of the last probe received from the neighbor. This number is used to verify that the packet reception information is fresh. This method of combining the feedback information with the probe packet minimizes the probing overhead. With the default value of the probing rate (N = 1, as listed in Section 6.4), the probing overhead in the vicinity of a node with 10 neighbors is approximately 1% of total airtime, and therefore does not significantly impact the throughput of MARS.

A mesh node receives periodic probe packets from each of its neighbors. The only probe packets of importance to a node on the tree are those from the node's upstream neighbor in each distribution tree. In other words, of all the possible links, MARS is only concerned with the PDRs of links that form the distribution tree. Therefore, the feedback in

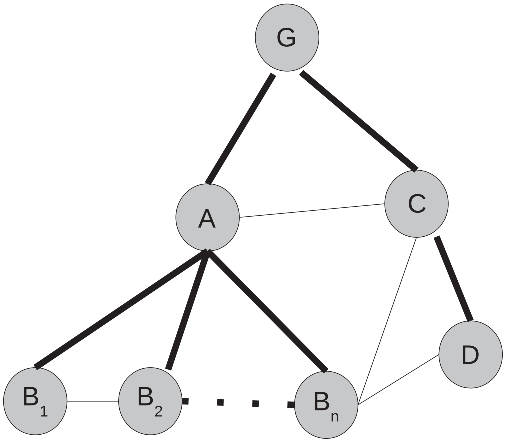

Fig. 2. Example mesh topology: node G is the gateway. Thick lines indicate links that form the multicast distribution tree.

the probe packets need only contain PDR statistics from the upstream neighbor(s) of the probing node. Locally collected PDR statistics from other neighbors can be discarded. Forwarder nodes in the distribution tree act as both upstream and downstream nodes. In the role of a downstream node, the forwarder node sends feedback to its upstream neighbor. In the role of an upstream node, the forwarder node collects feedback from its downstream neighbors.

As an example, consider the probe packets on a link AB_1, where A and B_1 are the upstream and downstream nodes, respectively, as shown in Fig. 2. On receiving a probe packet from A, node B_1 uses the first part of the probe packet to update its count of probes received from A at the bit-rate of the received packet. A probe packet later transmitted by B_1 contains the packet reception information from its upstream neighbor A. This part of the probe packet provides feedback to A regarding the PDR at each bit-rate. The feedback also includes LastSequenceNumber from A. The sequence number acts a virtual timestamp for A to verify that the PDR information is not stale. At node A, the feedback information from B_1 is stored in a decision table along with that from other downstream neighbors of A. This information is used for choosing a transmission bit-rate, as described in Section 4.4.

4.4. Choosing a transmission bit-rate

An upstream node A in the distribution tree receives feedback from all of its downstream neighbors. The feedback information from a downstream neighbor B is a list of the number of packets received at each bit-rate. For a multicast group M and upstream node A, let $B_1, B_2, \ldots, B_n$ be the n downstream neighbors that receive the multicast frame (see Fig. 2). For each multicast group M, we define a parameter β that represents the maximum tolerable MAC layer frame loss rate. We choose the bit-rate r such that, for all n downstream neighbors, r is the highest bit-rate with PDR greater than $(1 - \beta)$. In other words, the packet loss rate $(1 - PDR)$ of the individual downstream links at the chosen bit-rate r is less than β. The value of β may be chosen based on the application pertaining to the group (audio, video), the error recovery capabilities of the codec, or higher layer mechanisms such as reliable transport layer multicast or forward-error correction. For example a voice stream can sustain a packet loss rate of 10% assuming the presence of error concealment algorithms [29], whereas a 300 kbps video stream that uses the H.263 codec with appropriate error correction techniques suffers only a marginal reduction in quality in the presence of 15% packet loss rate [26]. The bit-rate selection decision is done periodically at a period of γ seconds. We experimentally study the impact of varying γ in Section 6.4.

Since multicast frames are not retransmitted, one of the goals of the bit-rate selection process is to maximize data delivery in the first transmission attempt. Metrics such as Expected Transmission Time (ETT) used in unicast rate selection [18] are not applicable to multicast bit-rate selection due to the absence of multicast retransmissions. PDR, closely related to data delivery, is therefore an appropriate metric for multicast bit-rate selection. Signal strength (or equivalently, SINR) of the links is an alternate metric for bit-rate selection, similar to the CHARM unicast rate selection algorithm [19]. However, accurate and consistent measurement of SINR in a heterogeneous mesh network environment, and translation of the measured SINR values to a data delivery metric, is non-trivial [19]. On the other hand, PDR can be measured directly with relatively low overhead, and therefore is used as the bit-rate selection metric for MARS.

4.5. Adding retransmissions to link-layer multicast

We now explore the idea of adding redundancy to the multicast frame transmissions through retransmissions at the MAC layer. The IEEE 802.11 protocol, as well as the basic MARS protocol, assume that each multicast frame is transmitted only once. On lossy links, this may lead to unsuitably high packet losses. To minimize the link-layer packet losses, we propose MARS-Retransmit (MARS-R), an extension to the base protocol MARS. The basic idea of MARS-R is to enable retransmission of multicast frames in the following manner: each upstream node nominates one of its downstream neighbors to respond to a successful frame reception with an ACK frame. We use the 4-address frame format of IEEE 802.11 instead of the usual 3-address format. The additional address field is populated with the address of the nominated downstream neighbor. We use the default IEEE 802.11 unicast retransmission behavior in terms of maximum retransmission attempts, random backoff intervals between consecutive transmission attempts. In other words, each multicast frame may be retransmitted up to eight times until the nominated receiver responds with an ACK frame. The frame is dropped if an ACK is not received even after all the retransmission attempts.

The probing and data collection component of MARS remains the same in MARS-R. The module for selection of the multicast transmission bit-rate is modified as follows. The ideology underlying this selection algorithm is to choose the bit-rate that requires the least time for successful reception of a packet, including the time for (re)transmissions and the corresponding MAC layer overhead. Consider the upstream node A, and its downstream neighbors $B_1, B_2, \ldots, B_n$ from Fig. 2. For each bit-rate r, node A calculates the Multicast Expected Transmission Time $METT_r = max(ETT_{B_1,r}, ETT_{B_2,r}, \ldots, ETT_{B_n,r})$, where $ETT_{B_k,r}$ is the unicast Expected Transmission Time for the link AB_k using bit-rate r for transmissions. The unicast ETT is calculated in a manner similar to that in SampleRate[3] [18]. For a given bit-rate, the unicast ETT value varies inversely as the PDR, i.e., on unreliable links more time is required for successful delivery of a packet. Further, for the same PDR value, unicast ETT varies inversely with the bit-rate. Node A selects bit-rate r_i with the least *METT* value for multicast transmissions. The downstream neighbor B_k with the highest ETT_{B_k,r_i} is chosen to respond with an ACK for multicast transmissions. In other words, the *METT* value for each bit-rate is determined by the link with lowest PDR at that bit-rate. Having selected bit-rate r_i with the least *METT* value, the least reliable link at bit-rate r_i is used for controlling the retransmissions.

The above algorithm for selecting a bit-rate with MARS-R is different from that of MARS since it targets reliability of multicast frame transmissions irrespective of the higher layer application. It is possible to adapt MARS-R to be application specific by modifying the above calculation of unicast ETT to be the expected time for a link PDR of $(1 - \beta)$. However, in this paper, we restrict ourselves to the basic ETT calculation for perfect reliability ($\beta = 0$).

4.6. Discussion

4.6.1. Probing

MARS probe packets cannot be combined with other probes (e.g. routing beacons) because MARS probes are at multiple bit-rates, whereas other probes are usually fixed bit-rate. However, it is possible to leverage the probing for unicast rate adaptation (e.g., SampleRate) to reduce the total overhead.

Currently, MARS probes all the possible bit-rates during each cycle. However, this probe traffic could be reduced with some heuristics similar to those in SampleRate [18]. For example, bit-rates with consistently low PDRs may be blacklisted and probed infrequently. Alternately, probing can use information from existing multicast traffic to enhance the probe data. However, this requires the sender/forwarder node to inform receivers about the number of frames transmitted at each bit-rate.

4.6.2. Rate selection interval

Previous research has suggested that rate adaptation for unicast transmissions should occur at short time-scales (of the order of a few packets) to react to short-term link quality variations [17,19]. A similar approach for multicast transmissions, however, is infeasible. Unlike unicast, multicast transmitters do not receive per-frame ACKs, and therefore, do not know immediately whether the packet was successfully received by all the downstream nodes. (Even in the case of MARS-R, the transmitter receives ACK from only one downstream neighbor.) This lack of immediate feedback forces the transmitter to rely on aggregate PDR information that is obtained based longer observation periods (of the order of a few seconds). Correspondingly, the bit-rate selection interval for multicast transmissions is also coarse-grained, compared to that for unicast transmissions. We further explore the bit-rate selection interval in Section 6.4 via experiments.

4.6.3. Reliability

Because MARS is a link-layer rate adaptation solution, it does not address the problem of reliability of multicast packets, which is typically handled at higher layers. Possible solutions for reliability include loss-resilient video codecs, forward-error-correction (FEC) based packetization, and unicast retransmissions of missing frames based on requests from receivers. The MARS-R scheme increases the probability of successful link-layer frame delivery via retransmissions, and contributes to the overall reliability of the stream. We note that the MARS-R scheme does not guarantee packet delivery to each downstream neighbor. Previous research has shown that packet loss events at neighboring locations may be independent of each other [30]. However, the nomination of the least reliable downstream link to respond with an ACK increases the number of retransmissions, and the resulting increase in redundancy of the system improves the reliability of packet reception.

4.6.4. Distributed bit-rate selection

One of the goals of multicast bit-rate selection in WMNs is to dynamically adapt the decision based on network topology and link conditions. An alternate approach would be to manually fix a single rate for all the nodes in the network based on link quality estimates [6,31]. Such a network-wide fixed rate would be influenced by the worst link in the topology, and cannot take advantage of regions of the network with good links. Further, such a solution cannot react to temporary changes in link qualities and therefore may not provide the best throughput performance. In contrast, the bit-rate selection of MARS is distributed in nature: each node uses local information to select the best bit-rate. The selection is periodically reevaluated and a new (lower or higher) bit-rate may be chosen as a result of a change in link PDRs.

5. Implementation

We implement our protocol for Linux systems using the MadWifi driver (v0.9.3.3). Our implementation of MARS consists of two components: modifications to the MadWifi driver, and user-level modules.

A first step in the implementation is to modify the driver to consult a rate adaptation module for multicast transmissions. This module is independent of the unicast rate adaptation module. MadWifi maintains a data structure to store information on a per-node basis. The MAC address of the node is used as an index into this table. For the

[3] ETT is the expected time required for successful reception of a packet, and is a function of Expected Transmission Count (ETX) for the link, the necessary MAC overheads such as DIFS, SIFS, and random backoff intervals. The ETX of a link is the inverse of its PDR.

multicast module, we take advantage of the standard Ethernet MAC address translation for IP multicast addresses [28]. Each multicast IP address is mapped to a distinct MAC address. These multicast MAC addresses are not assigned to any network interface card vendors, and are interpreted by all network stacks to be reserved for multicast transmissions. We instrument the driver to maintain a per-MAC address data structure for multicast addresses. This enables the driver to select different bit-rates as well as maintain transmit and receive statistics for each multicast group. User-level programs can set the bit-rate for individual multicast MAC addresses via the `/proc` interface. For the MARS-R algorithm, we modify the driver to enable ACKs for multicast frames. For multicast groups with the retransmit feature turned on, multicast frames are transmitted with the IEEE 802.11 4-address format, with the group address as the fourth address. The address of neighbor responsible for sending ACKs is written to the first address field. In essence, the multicast frame is encapsulated in a unicast frame destined to a chosen downstream neighbor. As a result, the radios on other downstream nodes have to be operated in promiscuous mode so that they can receive such unicast-encapsulated multicast frames. Our tests with the promiscuous mode of operation did not reveal any significant performance reduction, since the *unwanted packets* are dropped at the driver itself and not passed on to the IP layer. The ACK capability, and the MAC address of the neighbor responsible for the ACK can be programmed independently for each multicast group via the `/proc` interface. Note that these capabilities of the driver are independent of the multicast rate selection algorithm.

The user-level module for MARS is implemented in *perl* and performs the following functions. The module sends periodic probe packets to the immediate 1-hop neighborhood. For ease of implementation, we use a multicast MAC address dedicated for MARS probes instead of the default broadcast MAC address. This implementation allows us to control the bit-rate of the probe packets without affecting other operations that use MAC layer broadcasts, such as unicast routing. The module is subscribed to this multicast group, and receives and counts the probe packets from neighboring nodes at different bit-rates. The module reads the path information from the routing software to determine the upstream and downstream neighbors. Applications inform the module about the multicast group they are interested in and the downstream neighbors that are members of the group. Periodically, the user-level module calculates the PDR for each neighbor-rate combination. The module then determines the multicast bit-rate for each group registered by applications, and informs the MadWifi driver of this information. With the MARS-R algorithm, the user-level module also determines the downstream neighbor responsible for ACKs, and informs the driver of the neighbor's MAC address.

One notable issue that arose during the implementation is that the MadWifi driver (v0.9.3.3) assumes the presence of two antennas, and uses the antennas for receive diversity. However, some of the radios in our network are equipped with a single antenna. Unicast transmissions are not affected by this assumption since the driver intelligently chooses to use the same antenna on which frames were received. In the case of multicast/broadcast transmissions, the driver uses the two antennas in a round-robin fashion. This resulted in periodic loss of multicast frames. This behavior has recently been noted by other researchers as well [32]. To avoid this problem, we force the driver to use only one antenna for multicast transmissions from the single antenna radios.

6. Evaluation

We now evaluate the performance of MARS and MARS-R on the UCSB MeshNet testbed. We compare the performance of MARS and MARS-R against that of normal IEEE 802.11 with 1 Mbps as the bit-rate used for multicast/broadcast transmissions, and study the throughput gains achieved. Further, we evaluate resource consumption in the network in supporting the multicast streams. All the evaluations were conducted on the UCSB MeshNet testbed described in Section 3.2. The testbed uses the SRCR routing protocol for unicast traffic [27]. As described earlier, we use the tree formed by the unicast routes to form the multicast distribution tree.

For the purpose of evaluation we use a custom-built application module that handles the multicast group operations such as membership subscription, packet reception and forwarding. The application module resides at each mesh node. A node sends a unicast *join* message to the peer application on the upstream neighbor to subscribe to the multicast test application group 224.0.0.40. On receiving this message, the upstream node records the identity of the requester. At the gateway node, the application module acts as a CBR source that generates UDP packets for the application group. Each packet has a TTL of one to limit it to a 1-hop neighborhood at the MAC layer. At the other mesh nodes, the application module receives packets destined to the application group and logs them along with a timestamp. If the node is a forwarder, it queues the packet to be retransmitted. A simple sliding-window mechanism is used to track packet reception. Each receiver knows the CBR rate of the sender and uses packet sequence numbers to track a window equivalent to one second of the flow. Packets received with sequence numbers outside the sliding-window are discarded; duplicate packets are also discarded.

At the end of each experiment, the application at each node reports the average throughput, number of packets received, number of duplicate packets received, and average delay. From the MadWifi driver, we also record the number of frames received and transmitted at each bit-rate. Unless specified otherwise, we consider a scenario wherein all nodes in the network are members of the multicast group. This scenario represents a network-wide broadcast of the stream, and allows us to test the algorithm with a wide variety of link qualities. The default values for the parameters used are: bit-rate selection interval $\gamma = 60$ s, application sustainable packet loss rate $\beta = 15\%$, probes per second $N = 2$, and PDR feedback window duration $\lambda = 30$ s. We explore alternate values for these parameters in Section 6.4.

6.1. Performance

In this set of experiments we evaluate the goodput increase achieved by MARS and MARS-R. We use goodput (i.e., the rate of successful data reception) as the performance metric due to the lack of data delivery feedback at the upstream nodes as well as the importance of packet delivery ratio for multimedia applications. We conduct a series of experiment trials as follows. We vary the stream data rate from 64 kbps to 1536 kbps. The CBR source generates 1000 byte packets, and we vary the packet generation rate to control the offered load. Each trial lasts 300 s, and we repeat the experiment five times for a given offered load. The goodput calculated at each individual node considers only packets received successfully. Delayed packets and duplicate packets are excluded from this calculation. We measure the average goodput achieved for the group as the average of the individual goodput at each mesh node. We also plot the maximum and minimum goodput for each offered load. The experiment is repeated for MARS and MARS-R as well as basic 802.11 with 1 Mbps as the multicast bit-rate.

To isolate the impact of the network topology and environmental characteristics, we calculate the impact of the inherent link-layer loss in the network as follows. We measure the PDR for 1 Mbps probe packets during a 5-min*silent* period prior to each experiment. This allows us to estimate the native packet loss rate for the links in the network. This fraction of packet loss is a manifestation of the network topology, and not of MARS. The offered load multiplied by the PDR gives us a good estimate of the maximum achievable goodput given the loss characteristics of the network, assuming no link-layer retransmissions. Note that this estimate only considers the impact of PDR, and does not incorporate channel capacity constraints, i.e., it does not account for congestion caused by overloading the channel.

Fig. 3 plots the average goodput against the different offered loads for MARS and basic 802.11. We also plot the estimated maximum goodput for reference. We observe that the goodput for 802.11 peaks at about 256 kbps. On the other hand, MARS is able to provide goodput of up to 870 kbps. For offered loads of more than 1024 kbps, there is a drop in the goodput. This drop is caused by the increased packet generation rate, which leads to high contention and packet loss due to collisions. These collision losses also impact the probe traffic, and result in MARS choosing 1 Mbps bit-rate for transmissions. We note that the goodput for MARS is close to that of the estimated maximum until 1024 kbps offered load. MARS-R is able to provide a goodput of up to 982 kbps. With the help of retransmissions, MARS-R algorithm is able to provide higher goodput than the estimated maximum. This is because the redundancy in transmissions results in more nodes receiving the packet than with the default operation of single transmissions, thereby improving the PDR. This shows that retransmission of multicast frames helps in recovery from packet loss and is indeed helpful in better supporting multimedia streams. The goodput with MARS-R differs slightly from the offered load, i.e., it does not ensure 100% PDR. This is a manifestation of the phenomenon of mutually independent packet losses [30]. With offered load of more than 1152 kbps, the drop in goodput for MARS-R is steeper than for MARS. This is because the increased overhead of retransmissions adds to the existing network congestion and exacerbates it. We verify this by studying the fraction of transmissions constituted by retransmitted packets. For offered loads of less than 1152 kbps, retransmitted frames constituted less than 12% of all the transmissions; for higher loads they accounted for more than 23% of all transmissions.

To verify that the goodput gains are indeed from the use of bit-rates higher than 1 Mbps, we study the distribution of bit-rates used for transmissions. Fig. 4 plots a histogram for one representative trial of the experiment (offered load of 512 kbps). From the figure, we observe that MARS transmits about 80% of the frames using a bit-rate of either 5.5 Mbps or 11 Mbps. MARS-R transmits more than 90% of frames using bit-rates of 5.5 Mbps or more. In particular, about 8% of frames use either 24 Mbps or 36 Mbps. We also note that the algorithm prefers the non-OFDM bit-rates of 5.5 Mbps and 11 Mbps rather than the OFDM bit-rates of 6 Mbps and 9 Mbps. This reduced robustness (i.e., lower PDR) for OFDM bit-rates has also been observed by previous research [19].

The predominant use of 5.5 Mbps and 11 Mbps in this network suggests that an alternate method to improve

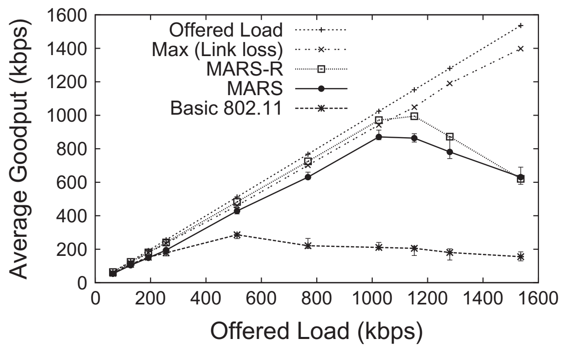

Fig. 3. Goodput performance in MeshNet.

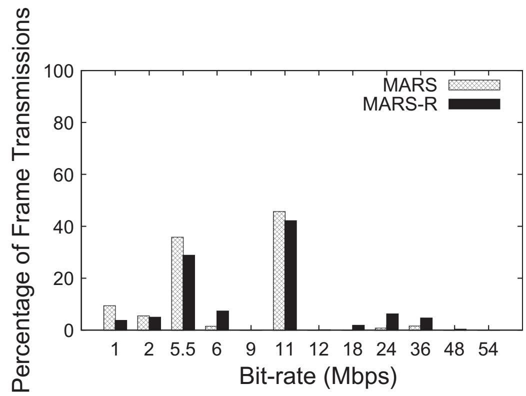

Fig. 4. Distribution of transmission bit-rates used by MARS and MARS-R in a representative experiment trial.

network performance for multicast is to use these higher bit-rates as the network-wide default rate. This method, however, has several disadvantages. First, the network administrator has to perform extensive measurements to determine the optimal rate for a given network topology. This optimal-rate in most cases is determined by the weakest link of the network and requires the use of the lowest bit-rate. Second, the use of the lowest common bit-rate does not take advantage of high quality links in the network that could have supported higher bit-rates. Third, the optimal bit-rate may be time-varying and the administrator may need to frequently tune the default rate. On the other hand, MARS is suitable for all network topologies as it dynamically selects the optimal bit-rate for each network, can take advantage of high-quality network links, and can vary this selection over time in response to changes in network conditions. In summary, the arguments for using dynamic multicast bit-rate selection instead of a fixed bit-rate are similar to those that motivate the use of rate adaptation for unicast transmissions.

6.2. Resource consumption

We now study the channel airtime consumed by multicast flows. Intuitively, a packet sent at a higher bit-rate requires less transmission time than when sent at a lower bit-rate. If a multicast flow uses the wireless channel for a smaller fraction of the time, the network can better support other traffic in the network with the remaining time. Note that these savings are in conjunction with the same or higher goodput as demonstrated in the previous section. To measure the resource consumption, we again refer to the experiments described in Section 6.1. Specifically we use the number of packets transmitted at each bit-rate, and the total number of received packets. We compute the metric of resource cost as the ratio of total transmission time per successfully delivered packet. Total transmission time is the sum of the time required to transmit packets from the entire flow at the gateway node and at each forwarder node. This number is divided by the total number of useful packets (i.e., excluding delayed and duplicate packets) received at all the nodes. The actual numerical value of this metric is a function of the network topology and distribution tree. Therefore we normalize the metric for MARS with respect to the corresponding metric for basic 802.11.

Table 1 lists the normalized resource cost metric at each offered load. A cost metric value of 0.2 means that MARS consumed 20% of the resources required by basic 802.11 to deliver a packet. From the table, we observe that the metric value varies from 0.16 to 0.387 for MARS and 0.253 to 0.462 for MARS-R. These values are influenced by the distribution of bit-rates used for transmissions, which in turn depends on the network topology. Together with our observations from Section 6.1, we conclude that MARS provides similar or higher throughput while consuming far fewer network resources. For example, MARS is able to sustain a 256 kbps stream for only 23% of the cost. The resource cost for MARS-R is marginally higher than that of MARS because the retransmissions of MARS-R increase the total transmission time, and at the same time increase the packet delivery rate for the stream.

Table 1
Resource cost.

Offered load (kbps)	Normalized resource cost	
	MARS	MARS-R
64	0.225	0.233
128	0.246	0.253
256	0.234	0.259
512	0.206	0.284
768	0.167	0.287
1024	0.243	0.311
1152	0.285	0.315
1280	0.346	0.383
1536	0.387	0.462

The reduced resource consumption for MARS is also reflected in the end-to-end delay for a packet. The delay for a multicast packet is defined as the time difference from when packet was transmitted by the source node to when the packet was last received by a node in the network [14]. For a 256 kbps stream, the average delay was 194.2 ms, 67.2 ms, and 79.4 ms for basic 802.11, MARS, and MARS-R respectively. We note that the overhead of retransmissions increases the delay for MARS-R compared to MARS.

6.3. Impact of packet size

In Section 6.1 we observed that with increased contention the average goodput drops sharply. We now perform a set of experiments to characterize the impact of packet size and packet generation rate. The experiments are similar to those described in Section 6.1. However, we fix the offered load to 512 kbps. We vary the packet size from 256 bytes to 1500 bytes, and to provide 512 kbps offered load, we vary the packet generation rate accordingly (from 500 packets-per-second (pps) to 42 pps). We repeat the experiment for MARS, MARS-R and basic 802.11 with 1 Mbps as the fixed multicast bit-rate. Fig. 5 shows the results of these experiments. With a small packet size of 256 bytes, the high number of packets, associated overhead, and increased contention results in very low goodput for MARS and MARS-R. The retransmissions of MARS-R also add to the contention, and we see lower goodput for MARS-R than

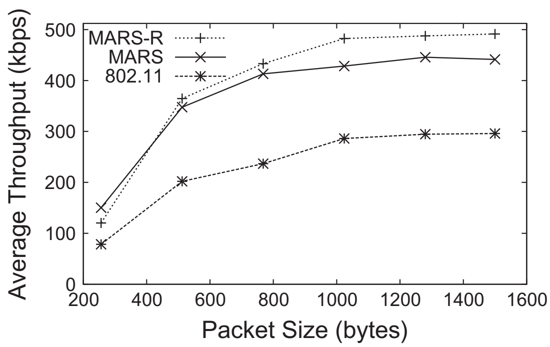

Fig. 5. Impact of packet size on MARS' throughput.

MARS in this case. The goodput increases with increasing packet size (and decreasing pps) because of the reduced number of collisions. The same experiment with 1 Mbps transmissions of 802.11 also shows similar trends of better throughput at higher packet sizes, albeit lower than the throughput of MARS. We conclude that the choice of packet size and packet generation rate is a critical factor in the achieved throughput with MARS and MARS-R. In fact, a repeat of the experiments in Section 6.1 with 1500 byte packets resulted in a MARS peak throughput of 1050 kbps (compared to 870 kbps at 1000 byte packets). Applications that generate small size packets can use frame aggregation techniques to obtain the benefits of using larger packets. Chen et al. suggest the joint-optimization of packet size and transmission bit-rate for unicast transmissions [33]. A similar approach for multicast packets is an avenue for future research.

6.4. Parameter selection

We now study the performance of MARS relative to the different parameters used. In particular, we vary the values of two parameters: bit-rate selection interval γ, and number of probes per second N. The offered load is 1024 kbps with 1000 byte packets.

First, we vary γ from 30 s to 180 s. During these experiments we fix $N = 2$. We fix the PDR feedback window duration λ to be 30 s, i.e., the probes have PDR information for the past 30 s. When $\gamma > \lambda$, MARS stores PDR information from previous probes to maintain data for the past γ duration. Table 2 lists the throughput achieved for each γ value. For rate selection intervals of 60–120 s, we observe high throughput. The throughput drops with an interval of 30 s and intervals greater than 120 s. At 30 s, MARS has a small number of probe samples to determine the bit-rate, and is therefore sensitive to occasional loss of probe packets. MARS becomes unresponsive to changes in network conditions when γ is more than 120 s. Both these scenarios lead to reduced throughput. We observe similar trends in throughput variation for MARS-R.

Next, we vary the number of probes per second (N). For these experiments, $\gamma = 60$ s and $\lambda = 30$ s. Table 3 lists the N values and the corresponding throughput for MARS and MARS-R. We observe that the throughput is sensitive to the value of N. There are two reasons for this behavior. First, a low value of N (<1) implies infrequent probing, and consequently MARS lacks a sufficient number of probe samples for bit-rate selection. Second, a high value of N (>4) leads to a large number of probe packets. These probes contend with the multicast data packets and increase the packet losses. Based on this tradeoff, we recommend a value of $N = 1$. Note that this rate of probing enables MARS to react to long-term (order of tens of seconds) variations in link quality rather than short-term (sub-second) variations. We note that the parameters γ and λ are uniform throughout the network. As part of future work, we intend to explore whether these parameters may vary in the network, e.g., based in number of neighbors.

Table 2
Impact of bit-rate selection interval on goodput.

Rate selection interval γ (s)	Goodput (kbps)	
	MARS	MARS-R
30	838.5	953.3
60	873.2	970.8
90	864.6	969.5
120	863.4	957.5
150	847.7	945.1
180	834.9	938.4

Table 3
Impact of number of probes per second.

Probes per second N	Goodput (kbps)	
	MARS	MARS-R
0.25	838.5	941.3
0.5	866.2	964.7
1	884.6	981.2
2	873.2	970.8
4	837.7	947.2
6	822.9	924.1
10	790.3	909.0

6.5. Impact of group topology

We now study the performance of MARS under different multicast group topologies. We randomly select nodes in the mesh network to be members of the multicast group. The remaining nodes in the network are not members of the group. However, depending on the location of the member nodes in the distribution tree, some of the non-member nodes may act as forwarding nodes. For goodput computation, we consider member nodes only. As before, the gateway node is the source of the multicast traffic. The experiment setup is similar to that described in Section 6.1. We find the maximum goodput for each topology.

We conduct this experiment for eight random topologies, and plot the maximum multicast throughput for each. The eight topologies have 9, 7, 10, 8, 8, 9, 12, and 7 member nodes, respectively. We conduct the experiment for MARS, MARS-R and default 802.11. Fig. 6 plots this result for the eight topologies. From the figure we see that the goodput of MARS and MARS-R is higher than that of basic 802.11 in all topologies. Not surprisingly, the magnitude of the

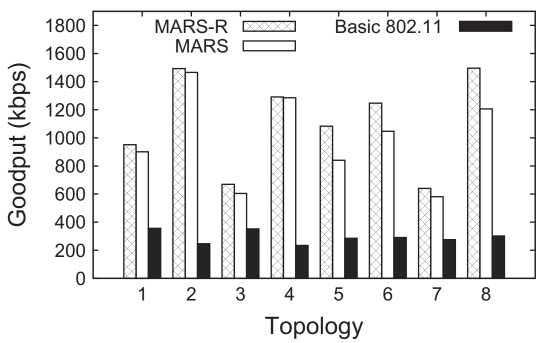

Fig. 6. Throughput performance for different topologies.

Table 4
Median bit-rate used in the different topologies.

Topology	1	2	3	4	5	6	7	8
MARS bit-rate (Mbps)	5.5	11	2	11	5.5	5.5	2	5.5
MARS-R bit-rate (Mbps)	5.5	11	2	11	11	5.5	2	11

gain is topology dependent. The improvement in goodput ranges from 171% to 595% for MARS and 190% to 606% for MARS-R. To better understand the result, we list the median bit-rate used for transmissions in each experiment in Table 4. We first focus on the performance gains of MARS over basic 802.11. Topologies where a large number of MARS transmissions can use high bit-rates result in high throughput. For example, the median bit-rate used by MARS for topology 4 is 11 Mbps. In contrast, the throughput improvement in some topologies (e.g., topologies 3 and 7) is less pronounced. The median bit-rate in both these topologies is 2 Mbps. A deeper inspection of the geographical distribution of bit-rates chosen by MARS provides insight into this behavior. The links to some nodes are of poor quality, and require the use of lower bit-rates for robust communication. When such a node is a member of the multicast group, the upstream neighbor is forced to use the lower bit-rate, and the overall throughput of the network is reduced.

We observe that the goodput for MARS-R is higher than that of MARS for all topologies. We attribute this to the increased data delivery via link-layer retransmissions. Two topologies, in particular, show significant (>200 kbps) increase in goodput. The median bit-rates for MARS-R in these topologies are higher than the corresponding bit-rate for MARS. This is because MARS-R can choose a higher bit-rate with low PDR (11 Mbps in this example) instead of a lower bit-rate (5.5 Mbps) with high PDR. The retransmissions allow MARS-R to compensate for the increased packet loss at high bit-rates. Since the time required to transmit a frame at a high bit-rate is reduced, the total transmission time for a packet (including time for retransmissions) may be less than the time at a lower bit-rate. A similar phenomenon is also observed in the unicast rate adaptation of SampleRate [18].

6.6. Joint operation of multiple groups

In this section, we conduct experiments to study the performance of MARS in the presence of multiple multicast groups. We show that MARS can take advantage of the different sets of downstream neighbors for different groups. We spawn three instances of the multicast application, each tuned to a different multicast group. We use topologies 1, 2, and 5 from the previous section for the three groups, respectively. Each stream uses 1000 byte packets, and the offered loads are 256 kbps, 512 kbps, and 128 kbps, respectively. We observe the average throughput of each stream as well as the median bit-rate used for transmission of packets from the streams. Table 5 lists the results from the experiment. We see that all three streams obtain high throughput. Further, MARS chooses an appropriate bit-rate for each group, e.g., 11 Mbps for topology 2.

Table 5
Simultaneous operation of multiple multicast groups.

Stream ID	Load (kbps)	Throughput (kbps)	Median bit-rate (Mbps)
1	256	236.53	5.5
2	512	476.16	11
3	128	121.86	5.5

7. Conclusion

Multimedia streaming applications require high throughput multicast flows with minimal packet loss. The basic 802.11 protocol, however, restricts multicast frames to be transmitted using low bit-rates, and thereby limits the capacity for multicast streams. Further, the lack of retransmissions affects the reliability of multicast streams. In this paper, we proposed an auto-rate algorithm for multicast frames in a wireless mesh network. The proposed algorithm, MARS, enables high throughput multicast streams for multimedia applications. The use of higher bit-rates reduces the channel time required for multicast streams, and better enables multicast streams to coexist with other traffic, unicast or multicast. The algorithm also provides an option to increase the reliability of multicast streams with the use of link-layer retransmissions in conjunction with higher bit-rates. The MARS algorithm takes advantage of topology distribution of stream receivers, and can be tuned as per the application requirements. MARS-R is able to provide high throughput and increased reliability while at the same time reducing the network resource consumption. We believe that this twofold advantage of MARS/MARS-R makes it very attractive for widespread adoption and improve the quality of service for multicast streaming applications.

References

[1] ITU-T IPTV Focus Group Proceedings, December 2008. <http://www.itu.int/ITU-T/IPTV>.
[2] E.M. Royer, C.E. Perkins, Multicast operation of the ad-hoc on-demand distance vector routing protcocol, in: Proc. of MobiCom, Seattle, WA, August 1999.
[3] S.-J. Lee, M. Gerla, C.-C. Chiang, On-demand multicast routing protocol, in: Proc. of WCNC, New Orleans, LA, September 1999.
[4] P. Sinha, R. Sivakumar, V. Bharghavan, MCEDAR: multicast core-extraction distributed ad hoc routing, in: Proc. of WCNC, New Orleans, LA, September 1999.
[5] J.G. Jetcheva, D.B. Johnson, Adaptive demand-driven multicast routing in multi-hop wireless ad hoc networks, in: Proc. of MobiHoc, Long Beach, CA, October 2001.
[6] S. Roy, D. Koutsonikolas, S. Das, Y.C. Hu, High-throughput multicast routing metrics in wireless mesh networks, in: Proc. of ICDCS, Lisboa, Portugal, July 2006.
[7] G. Marfia, P. Lutterotti, S. Eidenbenz, G. Pau, M. Gerla, FairCast fair multi-media streaming in ad hoc networks through local congestion control, in: Proc. of MSWiM, Vancouver, Canada, October 2008.
[8] S. Ni, Y. Tseng, Y. Chen, J. Sheu, The broadcast storm problem in a mobile ad hoc network, in: Proc. of MobiCom, Seattle, WA, August 1999.
[9] B. Williams, T. Camp, Comparison of broadcasting techniques for mobile ad hoc networks, in: Proc. of MobiHoc, Lausanne, Switzerland, 2002.

[10] P. Jacquet, A. Laouiti, P. Minet, L. Viennot, Performance of multipoint relaying in ad hoc mobile routing protocols, in: Proc. of IFIP Networking, Pisa, Italy, May 2002.
[11] H. Lim, C. Kim, Multicast tree construction and flooding in wireless ad hoc networks, in: Proc. of MSWiM, Boston, MA, August 2000.
[12] J.E. Wieselthier, G.D. Nguyen, A. Ephremides, Energy-efficient broadcast and multicast trees in wireless networks, Mobile Networks and Applications 7 (6) (2002) 481–492.
[13] R. Gandhi, S. Parthasarathy, A. Mishra, Minimizing broadcast latency and redundancy in ad hoc networks, in: Proc. of MobiHoc, Annapolis, MD, June 2003.
[14] C.T. Chou, A. Misra, J. Qadir, Low-latency broadcast in multirate wireless mesh networks, IEEE JSAC 24 (11) (2006) 2081–2091.
[15] C.T. Chou, B.H. Liu, A. Misra, Maximizing broadcast and multicast traffic load through link-rate diversity in wireless mesh networks, in: Proc. of WoWMoM, Helsinki, Finland, June 2007.
[16] A. Kamerman, L. Monteban, WaveLAN II: a high-performance wireless LAN for the unlicensed band, Bell Labs Technical Journal 2 (3) (1997) 118–133.
[17] G. Holland, N. Vaidya, P. Bahl, A rate-adaptive MAC protocol for multi-hop wireless networks, in: Proc. of MobiCom, Rome, Italy, July 2001.
[18] J. Bicket, Bit-rate Selection in Wireless Networks, Master's thesis, Massachusetts Institute of Technology, 2005.
[19] G. Judd, X. Wang, P. Steenkiste, Efficient channel-aware rate adaptation in dynamic environments, in: Proc. of MobiSys, Breckenridge, CO, June 2008.
[20] J. Kuri, S.K. Kasera, Reliable multicast in multi-access wireless LANs, Wireless Networks 7 (4) (2001) 359–369.
[21] J. Villalón, P. Cuenca, L. Orozco-Barbosa, Y. Seok, T. Turletti, Cross-layer architecture for adaptive video multicast streaming over multi-rate wireless LANs, IEEE JSAC 25 (4) (2007) 699–711.
[22] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, J. Crowcroft, XORs in the air: practical wireless network coding, in: Proc. of SIGCOMM, Pisa, Italy, September 2006.
[23] D. Aguayo, J. Bicket, S. Biswas, G. Judd, R. Morris, Link-level measurements from an 802.11b mesh network, in: Proc. of SIGCOMM, Portland, OR, August 2004.
[24] H. Lundgren, K. Ramachandran, E. Belding-Royer, K. Almeroth, M. Benny, A. Hewatt, A. Touma, A. Jardosh, Experiences from building and using the UCSB MeshNet Testbed, IEEE Wireless Communications Magazine, April 2006.
[25] K. Ramachandran, I. Sheriff, E. Belding, K. Almeroth, Routing stability in static wireless mesh networks, in: Proc. of PAM, Louvain-la-neuve, Belgium, April 2007.
[26] R. Zhang, S.L. Regunathan, K. Rose, Video coding with optimal inter/intra-mode switching for packet loss resilience, IEEE JSAC 18 (6) (2000) 966–976.
[27] J. Bicket, D. Aguayo, S. Biswas, R. Morris, Architecture and evaluation of an unplanned 802.11b mesh network, in: Proc. of MobiCom, Cologne, Germany, August 2005.
[28] S.E. Deering, Host extensions for IP multicasting, IETF RFC 1112, August 1989.
[29] H. Wu, Y. Liu, Q. Zhang, Z.-L. Zhang, SoftMAC: layer 2.5 collaborative MAC for multimedia support in multihop wireless networks, IEEE Transactions on Mobile Computing 6 (1) (2007) 12–25.
[30] A.K. Miu, H. Balakrishnan, C.E. Koksal, Improving loss resilience with multi-radio diversity in wireless networks, in: Proc. of MobiCom, Cologne, Germany, September 2005.
[31] S. Chachulski, M. Jennings, S. Katti, D. Katabi, Trading structure for randomness in wireless opportunistic routing, in: Proc. of SIGCOMM, Kyoto, Japan, August 2007.
[32] L. Verma, S. Kim, S. Choi, S.-J. Lee, Reliable, low overhead link quality estimation for 802.11 wireless mesh networks, in: Proc. of WiMESH, San Francisco, CA, June 2008.
[33] C.-C. Chen, H. Luo, E. Seo, N.H. Vaidya, X. Wang, Rate-adaptive framing for interfered wireless networks, in: Proc. of INFOCOM, Anchorage, AK, May 2007.

Prashanth Acharya received his B.E. degree in Computer Engineering from National Institute of Technology, Suratkal, India in 2002. From 2002 to 2004, he was with Texas Instruments, Bangalore, India as a Software Design Engineer. He is currently pursuing his Ph.D. in Computer Science at the University of California, Santa Barbara. He is a member of the Moment Research Lab directed by Elizabeth M. Belding-Royer. His research interests include Quality of Service, multimedia streaming and multi-radio wireless mesh networks.

Elizabeth M. Belding is a Professor in the Department of Computer Science at the University of California, Santa Barbara. Elizabeth's research focuses on mobile networking, specifically mesh networks, multimedia, monitoring, and solutions for networking in under-developed regions. She is the founder of the Mobility Management and Networking (MOMENT) Laboratory (http://moment.cs.ucsb.edu) at UCSB. Elizabeth is the author of over 80 papers related to mobile networking and has served on over 50 program committees for networking conferences. Elizabeth served as the TPC Co-Chair of ACM MobiCom 2005 and IEEE SECON 2005, and the TPC Co-Chair of ACM MobiHoc 2007. She also served on the editorial board for the IEEE Transactions on Mobile Computing. Elizabeth is the recipient of an NSF CAREER award, and a 2002 Technology Review 100 award, awarded to the world's top young investigators. See http://www.cs.ucsb.edu/~ebelding for further details.