HIMAC: High Throughput MAC Layer Multicasting over Time-Varying Channels

Ai Chen, Dongwook Lee, Gayathri Chandrasekaran, and Prasun Sinha
Department of Computer Science and Engineering, The Ohio State University
email: {chenai, leedon, chandrga, prasun}@cse.ohio-state.edu

***Abstract*— Efficient, scalable and robust multicasting support from the MAC layer is needed for meeting the demands of multicast based applications over WiFi and mesh networks. However, the IEEE 802.11 protocol has no specific mechanism for multicasting. It implements multicasting using broadcasting at the base transmission rate. We identify two fundamental reasons for performance limitations of this approach in presence of interference and realistic time-varying channels: (a)** *Channel-state Indifference:* **irrespective of the current quality of the channel to the receivers, the transmission always uses the base transmission rate; (b)** *Demand Ignorance:* **packets are transmitted by a node even if children in the multicast tree have received those packets by virtue of overhearing. We propose a solution for MAC layer multicasting called HIMAC that uses the following two mechanisms:** *Unary Channel Feedback (UCF)* **and** *Unary Negative Feedback (UNF)* **to respectively address the shortcomings of 802.11. Our study is supported by measurements in a testbed, and simulations. We observe that the end-to-end throughput of multicast sessions using MAODV can be increased by up to 74% while reducing the end-to-end latency by up to a factor of 56.**

I. Introduction

Wireless LANs and mesh networks based on the 802.11 technology are being rapidly deployed in public hotspots to provide ubiquitous coverage. Although mesh networks are relatively newer than infrastructured Wireless LANs, they are already operational in several cities including Philadelphia, Las Vegas, Taipei, and Urbana-Champaign (cuwireless.net). With increasing wireless data coverage and increasing capabilities of hand-held devices, multimedia streaming based applications are becoming critical to support. The success of iPOD and vPOD has already established the surprisingly high demand for live and stored streaming multimedia content. These applications can significantly benefit from multicasting support from the network.

Although multicasting has been well studied at routing and higher layers, MAC layer multicasting has not been well explored. The current 802.11 protocol achieves multicasting at the MAC layer using broadcasting, as there is no explicit mechanism for multicasting. Two inherent problems of this approach arising due to interference and time-varying channels are as follows: (a) *Channel-state Indifference:* Broadcasting uses the base transmission rate which may be much lower than the highest acceptable rate for the multicast neighbors. (b) *Demand Ignorance:* Recent enhancements to multicast protocols [1] use packet overhearing in the multicast tree. If the children nodes of a given node have received a packet, there is no demand for that packet from the children nodes. However, 802.11 transmits multicast packets regardless of their demand.

In this paper we propose HIMAC, a solution for efficient, scalable and robust multicasting at the MAC layer that addresses the above limitations of multicasting in 802.11 MAC. HIMAC uses two novel mechanisms namely, *Unary Channel Feedback (UCF)* and *Unary Negative Feedback (UNF)* to respectively address the limitations.

In HIMAC, the multicast sender first broadcasts an RTS packet. After receiving RTS, the receivers that have received the data packet respond with a UNF and other receivers respond with a UCF to inform the sender the state of the channel. If the sender only receives UNF, which means that no receiver needs this data packet, the sender simply drops the data packet. If the sender receives UCF, the sender will forward the data packet with the highest rate that the receivers can accept base on the duration of the longest UCF. Overlapping many encoded UCF packets can not destroy the information conveyed in the longest UCF. Hence, feedback for lower data rate is encoded using longer UCF.

MAC layer multicasting with time-varying data rates has not been studied before, although some multicasting solutions at the routing layer have accounted for links with different data rates [2]–[4]. Most of MAC layer multicast protocols focus on the reliablility metric. Kuri and Kasera [5] provided a reliable multicast protocol for WLANs, which is not suitable for ad hoc networks. BMW [6] proposes an approach for reliable multicasting using a round-robin approach that amortizes the cost of querying each node for ensuring reliability, but it can introduce arbitrary long latency for data packets. The BMMM [7] approach increases reliability of the MAC layer, but it is not fully scalable. Several approaches [8]–[10] use busy-tone on a separate channel to implement multicast reliability. but use of a secondary channel increases the hardware complexity. In [11], authors present an approach that tries to increase throughput by enhancing the resource utility in networks. This approach uses queue-lengths and estimates of the number of responding neighbors by measuring the busy tone power level, to determine whether to defer or continue with a multicast transmission. It has been evaluated primarily for single-hop scenarios. But, it can introduce arbitrary long latency for data packets and has a divergent behavior when the network traffic is heavy in multi-hop scenarios. Another problem of this approach is that it is difficult to accurately estimate the number of responding neighbors by measuring the power level because of fading in wireless links [12]. Time-varying channels and rate control has been studied by other researchers for unicast transmissions [13]–[15]. To the best of our knowledge, our approach is the first MAC layer multicasting solution that accounts for realistic time-varying channels and uses multiple rates supported by the physical layer.

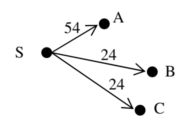
The organization of the rest of the paper is as follows. Section II presents the three limitations of 802.11 based MAC layer. Section III presents the complete design of HIMAC. Analysis of our protocol is presented in Section IV. The detailed performance evaluation using simulations are presented in Section V. Related work is discussed in Section VI. Finally, Section VII concludes the paper with pointers to future work.

II. Problems with 802.11 Multicast

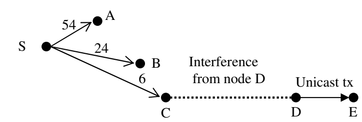
IEEE 802.11 implements multicasting by transmitting packets at the base transmission rate upon observing a clear channel. Unlike the RTS/CTS mechanism designed for unicast transmissions, it does not have any mechanism to obtain feedback from the intended multicast receivers. In this section we present two fundamental problems of multicasting using 802.11: *Channel-State Indifference and Demand Ignorance*, that justify the need for a new MAC layer approach for multicasting.

A. Channel-State Indifference

The properties of wireless channels are time-dependent due to factors such as interference, multi-path effects, and fading. As the channel properties are determined at the receiver, the sender needs to obtain feedback about the channel quality to each receiver in order to identify the best data rate to transmit multicast packets. As the multicast implementation in 802.11 is channel indifferent and does not use such feedback mechanisms, it transmits all packets at the base rate which is 6 Mbps for 802.11g/a and 1 Mbps for 802.11b. Consider an 802.11g/a MAC layer with one sender and 3 receivers as shown in Figure 1(a). The 802.11 protocol transmits packets at 6 Mbps. But, if the sender S can learn about the quality of the channel to the receivers, it can transmit at 24 Mbps, thus obtaining 4 times speedup in transmission time. If multicast packet transmissions at 54 Mbps are feasible considering the link qualities of all the receivers, the speedup is by a factor of 9. [1] Of course, the overheads of feedback communication, back-offs, and physical layer header (always transmitted at base-rate) will have to be accounted for, in order to obtain the net performance gain.



(a) No interference



(b) With interference

Fig. 1. Channel-State Indifference: 802.11 always transmits at the base-rate. (a) 24 Mbps multicast transmission rate is the best. (b) 24 Mbps multicast transmission rate is the best if the packet transmission from D to E interferes with C's multicast transmission.

If the receiver with the poorest channel can support a maximum rate of 6 Mbps then the sender needs to transmit at 6 Mbps. It may seem that in such scenarios, the channel feedback will be wasteful. However consider the example in Figure 1(b) which shows three receivers with maximum supportable rates of 54, 24, 6 Mbps. Suppose that an ongoing transmission on another link causes interference for user C with 6 Mbps channel to result in a collision with the multicast transmission. If the sender can learn about it in advance, it can still transmit at 24 Mbps maintaining a speedup factor of 4, without affecting the delivery ratio.[2]

To understand the potential benefit that can be obtained by channel-aware transmissions, we conducted an experiment using two laptops equipped with 802.11b NICs. The experiment was conducted in a long indoor corridor. The NIC used for the experiments was Netgear WG511T that is based on the Atheros chipset. As the NIC and the madwifi driver does not support rate adjustments for broadcast packets, we conducted our

[1]Transmitting at higher data rates always increases the error rate. But for good channel conditions, the increase in error rate may be acceptable. Most researchers [14] use 1% packet error rate as the guideline for determining the best data rate for unicast packets.

[2]If the sender needs to confirm all of 3 receivers receive the data packet, it is better to let the sender sends the data packet to nodes A and B first and sends it to node C later when there is no interference because it is hard to make every receiver ready to receive the packet in wireless environment especially when the traffic is heavy.

study using unicast packets. RTS/CTS and retransmissions were disabled to obtain measurements relevant for multicasting. Each point in Figure 2 is obtained by transmitting 10000 packets from one node to the other. The graph shows that when all receivers are located within 225 feet from the transmitting node, transmissions at 11 Mbps performs the best. Surprisingly in this experiment we observe that transmissions at 1 Mbps performs the best only for distances beyond 275 ft. These results show that channel-state awareness can significantly improve performance.

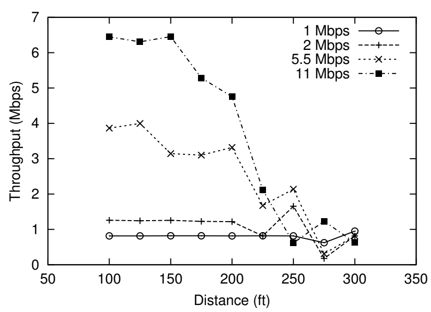


Fig. 2. Experimented throughput between two nodes in a corridor: 11 Mbps is best for a considerable distance. 1 Mbps is the best choice only beyond 275 ft.

Although the MAC layer can be modified to obtain feedback from all the receivers, the challenge is to do so in a scalable fashion such that the total time for communication is independent of the number of receivers.

B. Demand Ignorance

Most ad-hoc multicast protocols construct a tree [16] or mesh based sub-structure [17] in the network to forward multicast packets. However, recent advances in AODV [1] have suggested a solution involving packet overhearing to supplement forwarding losses on the tree. By taking advantage of the broadcast nature of the channel, this approach improves packet delivery ratio as nodes on the multicast tree can overhear multicast packets from other on-tree nodes in addition to receiving from its parent. This improvement in MAODV, brings out another shortcoming of 802.11 based multicasting. Although the sender may know the list of its children in the tree, it does not know whether any children is still missing a multicast packet or not. Thus the multicast packet transmitting node is ignorant of the demand for the packet. If all the children have received a packet by virtue of overhearing, then the multicast transmission may be wasteful.

Consider the example shown in Figure 3 which shows an MAODV session with sender S, receivers A, B and a forwarding node F. If a packet transmitted from S is overheard by all nodes, then for that packet the multicast transmission from node F is not needed. However, in absence of feedback from nodes A and B, F will end up transmitting the packet unnecessarily.

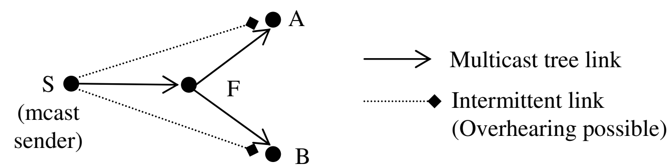


Fig. 3. Demand Ignorance: In absence of feedback, F can not know whether its downstream nodes have received the packet by virtue of overhearing or not.

Thus, feedback from the receivers regarding the status (received before or not) of the current packet will help the sender in avoiding unnecessary transmissions. Implementing this approach in a scalable way is challenging. Observe that lack of feedback can not be used as an indication for lack of need for the packet, as feedback may also be suppressed by the receivers due to interference from hidden terminals.

III. Protocol Description

HIMAC consists of two mechanisms namely *Unary Channel Feedback (UCF)* and *Unary Negative Feedback (UNF)* to address the limitations of multicasting in 802.11. We also present the implementation of unary feedback in the physical layer in this section.

A. Unary Channel Feedback (UCF)

The UCF mechanism addresses the Channel-State Indifference problem. The state of the channel is known only at a receiver, which can rapidly vary over short durations [14]. In order for the sender to choose the optimal data rate for transmitting multicast packets, it needs to learn the current state of the channel at all the receivers. A naive approach of obtaining separate feedback from each receiver just before transmitting the packet does not scale with the number of neighboring receivers. In addition, the delay and overhead of multiple feedbacks from receivers can easily outweigh any possible savings in data transmission time.

In order to ensure scalability and low overhead, HIMAC uses unary encoding of feedback from the receivers. In unary encoding, the duration is used to encode information. The sender first broadcasts an RTS (Request To Send) packet. The RTS packet will contain the multicast IP address to enable the receivers to determine if they need to respond with a UCF. This requires a cross-layer approach where the driver/firmware running the MAC protocol has access to the multicast routing tables.[3] In response, each potential receiver simultaneously sends a unary signal (UCF) that encodes the highest acceptable rate based on the channel quality measured after receiving the RTS. Because the rate selection is located on each receiver and just prior to data packet transmission, the selected rate is always suitable to data packet transmission for each receiver. As opposed to binary encoded packets that are susceptible to collisions, multiple unary signals will sum up to a single unary signal. We assume that the receivers of unary signal can determine the length of the longest unary signal even after multiple unary signals are summed up. This is possible even in cases with destructive interference as long as the sender can hear some activity in the channel near the end of the longest unary signal. The highest data rate is encoded with a short duration unary signal and progressively lower rates are encoded with longer unary signals. Therefore the summed up unary response at the sender is sufficient to determine the optimal data rate for transmission. We assume the sender always knows the longest length of the UCF responses if it receives the UCF responses, *i.e.* it is impossible that the sender gets the shorter UCF response and losses the longer UCF response. The reason is that the useful information of the UCF is just its length. If the sender gets the shorter UCF response and losses the longer UCF response, the longer UCF response will be as the interference of the shorter UCF response. The sender can not decode the length of the shorter UCF response with the interference of the longer UCF response. The absence of any UCF after sending the RTS indicates that the receivers (if any are still present in the neighborhood) will not receive the packet. Thus, when UCF is absent, unnecessary data transmissions can be avoided to improve performance. The sender re-attempts to transmit the packet after backing off according to unicast transmission rules in 802.11.

Figure 4 shows an example where the maximum possible rate is inferred from the longest UCF signal. In this example there are two receivers with channel rates of 36 Mbps and 24 Mbps. As the UCF for the 24 Mbps is longer than the UCF from the 36 Mbps receiver, the sender will learn that it needs to transmit the packet at 24 Mbps.

[3]If using a mac-layer multicast address is possible (every multicast IP address is map to a mac-layer multicast address), this cross-layer design is unnecessary.

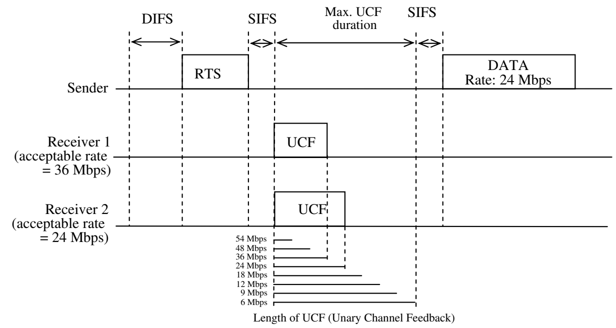


Fig. 4. Unary Channel Feedback: Multiple UCF signals from receivers will overlap but the sender can still determine the highest required rate based on the longest UCF signal.

Although a separate channel will be ideal for the UCF packets, it requires two radios and increases the hardware complexity. Instead, we assume that the UCF packets are sent in-band. A busy-tone at the carrier frequency of the channel is one possible implementation of the unary signal.

The UCF mechanism introduces overhead (RTS and UCF packets) for multicast in the MAC layer. But it provides the following benefits:

- The sender can confirm there is at least one receiver ready to receive the packet, which improves the packet reliability.
- The sender can use higher rate than the base rate to send data packets, which reduces latency.

For example, if the size of the data packet is 1000 bytes, the packet transmission time is 1333 μs at the base rate of 6 Mbps in 802.11g/a. The transmission time is 889 μs at 9 Mbps (the next higher data rate) and 148 μs at 54 Mbps. As the overhead of $RTS + UCF$ is only 94 μs (including 20μs of 2 SIFS intervals), transmission at any rate higher than 6 Mbps will improve performance.

B. Unary Negative Feedback (UNF)

In the UCF approach, the receivers respond with the UCF packet even if they have overheard that data packet before. We now present our solution to address this problem named as the Demand Ignorance problem. A simple extension of UCF can cache recent data packets[4] and avoid transmitting the UCF in case the packet has been overheard. This requires the maintenance of a cache that is indexed by packet IDs. Although locally unique

[4]Because the overheard data packet is sent by a neighboring node of the receiver and this node should not be far away from the receiver in the multicast tree, The receiver does not need to cache the data packet for a long time. Therefore, It is enough for the storage to keep a cache of several overheard packets, *i.e.* $10k\,bytes$.

packet IDs can be used for such purposes, a naive approach is to use a globally unique packet ID formed by the IP address of the source and a unique sequence number assigned at the source to identify the packets. The RTS will also contain the packet ID. Although this extension of UCF saves network resources by avoiding unnecessary feedback from some receivers, it is not sufficient to identify if all receivers have received the packet. In such a scenario, an absence of UCF will lead to multiple retransmission attempts by the sender till it exceeds the maximum limit for retries. This is wasteful as retransmissions are preceeded by backoff periods that increases exponentially with each attempt.

The UNF approach attempts to differentiate the scenarios of no demand for a packet and heavy interference. In the former case, the packet needs to be dropped whereas in the latter case the packet transmission is reattempted. In this approach, the receivers who have previously overheard the packet respond with a single unary-digit long signal to inform the sender. As with UCF, multiple UNFs may overlap without loss of any relevant information. The sender needs to learn the presence or absence of the unary-digit in case there is no UCF. The UNF signal is used by the sender only if there are no UCF transmissions. In such a case, the presence of UNF indicates that the receivers that heard the RTS and that have a clear channel to receive this transmission have already received the packet. Thus, the absence of UCF and the presence of UNF triggers the sender to drop the packet. There is still a possibility due to interference that only UNF is sent by some receivers, even when other receivers have not received the packet. The absence of both UCF and UNF strongly indicates that the receivers (if any are present in the neighborhood) are not in a position to receive the packet. In this case, the sender decides to retransmit RTS. Note that UNF approach is only useful in multi-hop wireless scenarios, where overhearing is possible. However, UCF is useful in both multi-hop wireless scenarios and one-hop wireless LANs.

Figure 5 shows an example where the two neighbors of a sender have received the packet by overhearing. On receiving the RTS, the receivers respond with a UNF. As both the receivers send the UNF, the sender sees no UCF and decides to drop the packet.

When UNF approach is implement, it is possible that the nodes who do have the data packet receive the RTS and send UNF, while the nodes who do not have the data packet do not receive the RTS and do not send UCF, which results in no data packet transmission. However, the nodes who do not have the data packet still can not

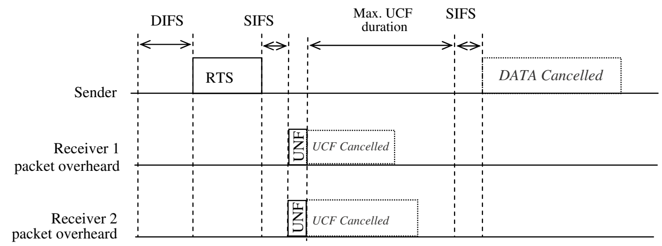


Fig. 5. Unary Negative Feedback: Receivers send UNF if they have overheard the packet before. The overlapping UNF signals still carry the message to the sender that all those nodes that can receive the packet do not have demand for this packet any more.

receive the data packet if the sender directly sends the data packet because these nodes can not receives the RTS when the sender sends the RTS first. Because the transmission time for RTS and UNF is much shorter than the transmission time for the data packet, HIMAC at least save the channel resource.

Combine UCF and UNF, HIMAC works as follows.

1) When the sender has the multicast data packet to send. the sender send the RTS to the receivers.
2) After receive the RTS from the sender, the receivers who have the data packet send UNF and the receivers who do not have the data packet send UCF to tell the sender the highest acceptable rates.
3) If the sender only receives UNF, the sender drop the data packet and the data transmission is canceled; if the sender receives UCF, the sender sends the data packets with the highest acceptable rates for all receivers who have sent UCF; the sender retransmit the RTS and the process goes back to 2 if the sender does not receive both UNF and UCF, and the number of retransmissions is less than the limitation same as the limitation for the unicast transmission.
4) After the receivers receive the data packet, the transmission is finished. There is no ACK sent by the receivers.

Because there is no ACK, the NAV set in the RTS is the time before the data packet transmission.

C. Implementation of Unary Feedback

The unary signals should enable the sender to reliably compute the minimum of the data rates of channels to the receivers. A naive implementation is to use the baseband signal to send a tone of the desired duration. However depending on the channel properties such as attenuation, multi-path effects and phase, multiple such overlapping signals may cancel each other, making it difficult for the receiver to decode the combined unary signal. We discuss two approaches based on current physical layer technologies for robust implementation of the concept of unary signals.

PN-codes: An approach to make the overlapping feedback more robust is to use Pseudo Random Noise Codes (PN-codes) [18]. Each receiver is assigned a unique PN-code. The sender can decode the length of the PN-coded signal corresponding to each receiver to determine the length of the feedback from each receiver. However, PN-codes require power control for optimal operation to address the near-far effect [18]. Closed-loop power control will incur additional signaling overheads.

OFDM sub-carriers: The current OFDM based 802.11 a/g hardware can also be leveraged to implement the unary signals. OFDM uses 48 sub-carriers to deliver data and 4 sub-carriers to send reference signals in every channel as shown in Figure 6. OFDM receivers can decode 52 sub-carriers simultaneously. Thus, if we assign different sub-carriers to different receivers, then simultaneous feedback can be obtained from all the receivers. If each receiver also needs to send one or more reference signals, then the number of supportable next hop neighbors will be reduced. If each receiver also sends 2 reference signals on two sub-carriers along with one sub-carrier for the unary signal, then 52/3 = 17 receivers can be supported using this solution. For designing a system that supports more receivers, users can be grouped into different slots where up to 17 receivers can use one slot. However, for a very large number of receivers, most transmissions will typically require the base-rate and our approach will have limited advantage over 802.11 which uses transmission at the base

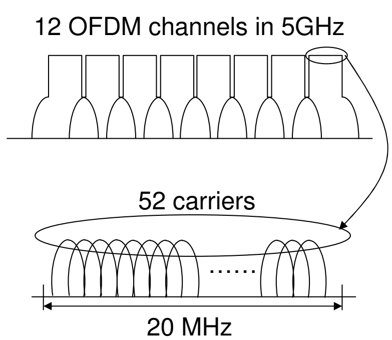


Fig. 6. IEEE 802.11a OFDM PHY channel.

An alternate OFDM based implementation is to assign a group of sub-carriers to one data rate. There are 8 transmission rates in IEEE 802.11a/g. Each transmission rate can be assigned 6 sub-carriers. Each user randomly chooses one of the 6 sub-carriers from the group corresponding to its data rate. For further scaling the system, users can either be grouped in time to different slots or assigned a random transmission time in a given interval for sending the feedback to the sender. Note that in this encoding, the feedback is implicit in the sub-carrier that is decoded at the receiver. So, the length of the unary signal has no information. The length should be enough to enable robust decoding. Unlike the other approaches, this approach has a finite probability that the feedback signal from multiple neighbors may collide.

IV. Protocol Analysis

HIMAC uses current information about the channels to the receivers to determine its actions. However the overhead introduced by the extra control packets can reduce the benefits of HIMAC. In this section, we analyze the performance of the UCF mechanism using some observations from the measurements presented in Figure 2.

We assume that m receivers are uniformly randomly distributed around a single sender. The physical layer supports n different transmission rates: $\rho_1 < \rho_2 < \rho_3 < \cdots < \rho_n$. We assume that corresponding to each ρ_i, there is a circular transmission range R_i ($R_1 > R_2 > R_3 > \cdots > R_n$) within which that data transmission rate results in an acceptable packet error rate.[5] A data rate ρ_i is said to be *feasible* for a user if the user is within a distance of R_i from the sender.

Let P_{ij} represent the probability that rate ρ_i is feasible for j users randomly placed around the sender. For a single receiver it is same as the probability with which the user will lie in the circle of radius R_i. Therefore,

$$P_{i1} = \frac{R_i^2}{R_1^2} \tag{1}$$

The probability P_{im} that the rate ρ_i is feasible for all the m receivers is:

$$P_{im} = P_{i1}^m = \left(\frac{R_i^2}{R_1^2}\right)^m \tag{2}$$

In the indoor experimental measurements we have observed that for a distance up to 275 ft, transmission at 2 Mbps results in higher throughput in comparison to transmission at the base rate of 1 Mbps. This implies that our UCF approach will improve performance as long as all receivers are withing 275 ft from the sender.

[5] We assume it is true to simplify our analysis although the packet error rate is also related to some other factors, such as interference.

The transmission range of the sender is about 300 ft. By using these ranges and Equation 2, we can compute the probability that the feasible rate is higher than the base rate. Figure 7 shows this probability for varying number of receivers. In our simulations we observe a typical value of 4 or 5 neighboring receivers. For 5 receivers, we observe that there is still a 40% chance that the transmission rate can be enhanced using HIMAC.

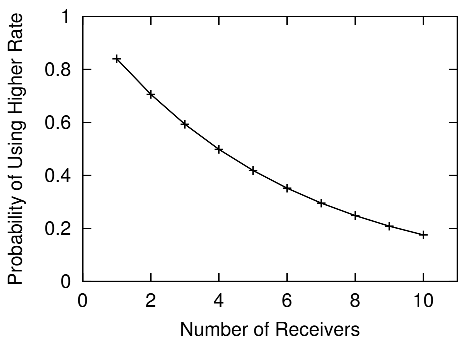


Fig. 7. Probability that feasible rate is higher than base-rate based on measurement data (Figure 2): The probability is still 50% when the number of receivers is 4; 18% when the number of the receivers is 10.

In HIMAC, the sender always uses the highest rate that it can use to send packets. Therefore, the expected transmission rate E_ρ for m receivers is (assume $P_{n+1,m} = R_{n+1} = 0$):

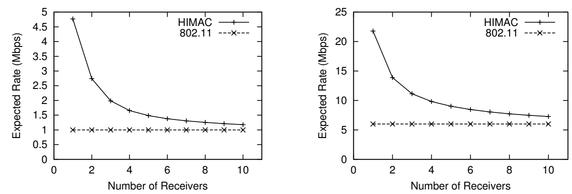
$$E_\rho = \sum_{i=1}^{n} \rho_i * (P_{im} - P_{i+1,m}) \tag{3}$$

$$= \sum_{i=1}^{n} \rho_i * \left(\left(\frac{R_i^2}{R_1^2}\right)^m - \left(\frac{R_{i+1}^2}{R_1^2}\right)^m \right) \tag{4}$$

We now compute the expected transmission rate for the two classes of 802.11 physical layers.

- *802.11b PHY:* In order to simplify calculating the expected rate for 802.11b, we conservatively assume the transmission range is 150 ft, 200 ft, 275 ft, and 300 ft for transmission rates of 11 Mbps, 5.5 Mbps, 2 Mbps and 1 Mbps, respectively. Using Equation 4, we obtain Figure 8(a), which shows the expected rate. When the number of the receivers is 5, the excepted rate is 1.48 Mbps. If the size of the data packets is 1000 bytes and there are 5 receivers, the expected transmission latency for a data packet is $5284\,\mu s$ [6] while it is $8000\,\mu s$ if the base rate of 1 Mbps is used. The latency of RTS and UCF (including $20\,\mu s$ SIFS delay) is $460\,\mu s$. Thus, even after taking into account the overheads of the RTS and UCF packets, HIMAC performs better than 802.11b.

[6]This calculation ignores the fact that the preamble and the PHY headers are always sent at the base rate



(a) 802.11b (b) 802.11a/g

Fig. 8. Expected Rate vs. Number of Receivers: (a) 802.11b: The expected rate is 1.66 Mbps when the number of the receivers is 4; 1.18 Mbps when the number of the receivers is 10. (b) 802.11a/g: The expected rate is 9.82 Mbps when the number of the receivers is 4; 7.27 Mbps when the number of the receivers is 10.

- *802.11a/g PHY:* Using the range ratios in Table II and Equation 4, we get Figure 8(b). For 5 receivers, the excepted rate is 9.01 Mbps. If the size of the data packets is 1000 bytes and there are 5 receivers, the expected transmission latency for a data packet is $887\,\mu s$ while it is $1333\,\mu s$ if the base rate of 6 Mbps is used. The latency of RTS and UCF (including $20\,\mu s$ SIFS delay) is $94\,\mu s$. Thus, even for 802.11a/g physical layers, HIMAC performs better.

V. Performance Evaluation

We have implemented our protocol in ns-2.28. Using extensive simulations we observe that HIMAC performs significantly better than 802.11. We use MAODV [16] as the network-layer multicast protocol. It should be mentioned that HIMAC is independent of the network-layer multicast protocol. HIMAC reduces the one-hop MAC layer latency by up to a factor of 6. The end-to-end throughput of multicast sessions using MAODV can be increased by up to 74% while reducing the end-to-end latency by up to a factor of 56.

A. Simulation Environment

Table I summarizes the simulation settings. These simulation settings are used for both 802.11 and HIMAC unless mentioned otherwise. Table II shows the transmission rates that the physical layer can support, the corresponding SNR thresholds, and the ratio of the corresponding transmission range to the range for the base rate. For modeling time-varying channels we implemented the Rayleigh fading model [12] in ns-2. Because of fading, the receiving power changes dynamically even when the transmission power and the distance between the sender and the receiver are fixed. The average transmission range corresponding to the base rate is about 150 meters. For every simulation case, 4 random scenarios have been simulated and each scenario's simulation time is 400 seconds. The minimum and maximum values are represented using vertical bars in the graphs.

Area Size	500×500 m^2
Number of Nodes	50
Maximal Node Speed	10 m/s
Number of Senders	1
Number of Receivers	10
DATA Packet Size	1460 bytes
Load	200 packets/sender/s
Transmission Power	0.28183815 W
Carrier-Sensing Power Threshold	2.35729217e-10 W
Receiving Power Threshold	6.041482e-09 W
CS/RX range ratio	2.25
MIN (basic) transmission rate	6 Mbps
MAX transmission rate	54 Mbps

TABLE I
SIMULATION SETTINGS: DEFAULT ENVIRONMENT VALUES UNLESS MENTIONED OTHERWISE.

The metrics used for performance study are as follows:

- **End-to-end throughput**: the average number of the packets received by each receiver per second.
- **One-hop MAC-layer latency**: the average latency of the receivers receiving the packets at the MAC layer.
- **End-to-end latency**: the average latency of the receivers receiving the packets at the network layer.

It should be mentioned that the end-to-end latency is not just the summation of the MAC-layer latencies. It includes other delays such as queuing delay. In the simulation results discussion, *Throughput* means end-to-end throughput, *MAC latency* means MAC-layer (one hop) latency, and *Latency* means end-to-end latency.

B. Network Density

By changing the network area, we study the impact of the network density. Figure 9 shows the results for different area sizes. From these figures, we observe that HIMAC (UCF and UNF mechanisms together) performs better than UCF. When the area is small, the throughput is high for all the three protocols as the receivers are close to the sender which increases the chances of receiving the packets. When the area is $100\,m \times 100\,m$, the latency of HIMAC is 0.006 seconds while the latency of 802.11 is 0.336 seconds. This can be explained by observing that when the density is very high, most of the nodes in the network share the same medium. Higher transmission rates used in HIMAC and UCF reduce the medium access time for transmissions. So HIMAC and UCF can still withstand the high load of 200 packets/seconds set in these simulation. For 802.11, this load is not manageable due to the use of base transmission rate. The impact is seen more on the delay experienced by the packets transmitted using 802.11. This is because every sender has to wait longer for accessing the medium, which increases the MAC latency and the queuing delay significantly. As UNF improves performance over UCF, the rest of the simulations in this paper compare 802.11 with HIMAC, which includes both the mechanisms.

C. Node Speed

Figure 10 shows the impact of the maximum speed for the random way-point mobility model. Figure 10(a) shows that the throughput is highest when the node speed is 1 m/s as Rayleigh fading is changed slowly. Slow speed also makes the multicast tree more stable. At all speeds, HIMAC has much higher throughput than 802.11. For low speed of 1 m/s, the throughput of HIMAC is 101.8 packets/receiver/second, while for 802.11 it is 65.6 packets/receiver/second. The improvement in throughput is 55.2%. HIMAC's latency and MAC latency are also much smaller than 802.11. HIMAC reduces MAC latency by 40% and latency by $10-20\%$

D. Packet Size

Figure 11 shows the impact of packet size. It is easy to see from the figures that HIMAC still outperforms 802.11 MAC even when the packet size is small. It implies that the benefit of the reliability and high delivery rate of HIMAC overcomes the cost of the control overhead related to HIMAC even for small packet sizes. When the packet size is 250 bytes, the MAC latency and latency of HIMAC are similar to the ones of 802.11. The throughput of HIMAC is higher than 802.11 by 10% because HIMAC improves the multicast reliability and uses multi-rate multicast in MAC layer. When the packets size is 125 bytes, the performance of HIMAC and 802.11 are almost same while the throughput of HIMAC is still a little higher than 802.11.

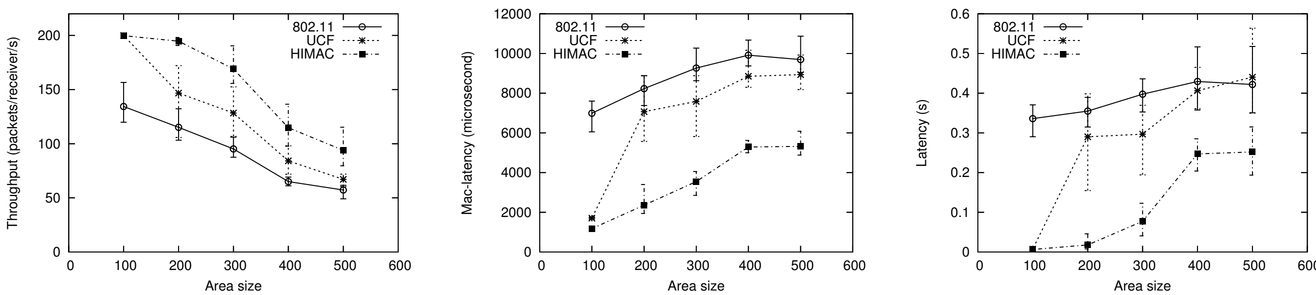
E. Network Load

Figure 12 shows the impact of load. We find that the throughput and latency of HIMAC is similar to 802.11 at low network loads, because the network bandwidth is high enough to deliver those packets to receivers, even using 802.11. The MAC latency of HIMAC is

Rate (Mbps)	6	9	12	18	24	36	48	54
SNR Threshold (db)	21	22	23	26	30	34	38	40
Trans. range ratio	1	0.94	0.89	0.75	0.60	0.47	0.38	0.34

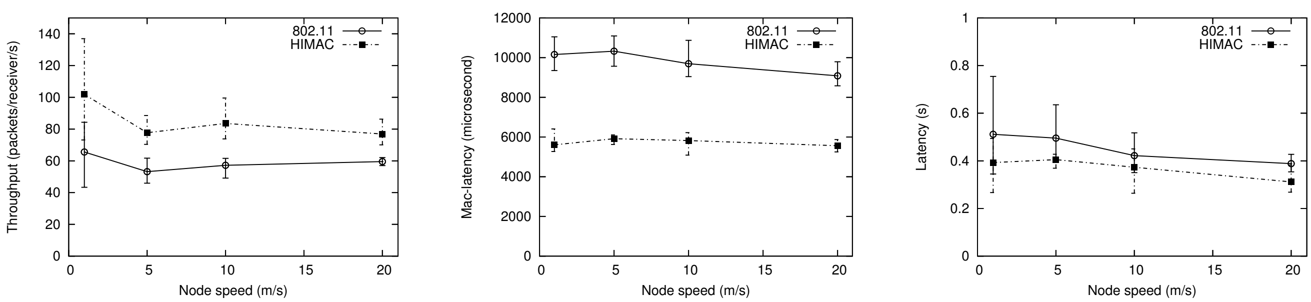
TABLE II

SNR THRESHOLDS AND TRANSMISSION RANGE RATIOS TO THE BASE RATE



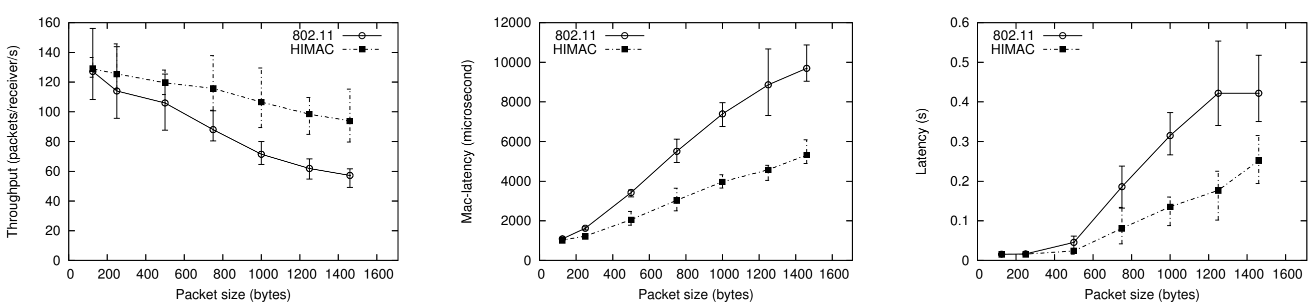
(a) Throughput (b) MAC Latency (c) Latency

Fig. 9. Network Density: When the area size is $100\,m \times 100\,m$, the latency of HIMAC is 56 times lower than 802.11 because of significant MAC latency and queuing latency reduction.



(a) Throughput (b) MAC Latency (c) Latency

Fig. 10. Node Speed: When the speed is 1 m/s, the throughput is highest and the improvement of HIMAC in throughput is 55.2% at a speed of 1 m/s. HIMAC reduces MAC latency by 40%.



(a) Throughput (b) MAC Latency (c) Latency

Fig. 11. Packet Size: When the packet size is 250 bytes, the throughput of HIMAC is higher than the one of 802.11 by 10% although their MAC latency and latency are similar.

always smaller than 802.11 because of high delivery rate of HIMAC. With increasing network load, HIMAC's performance improvement becomes more significant. The reason is that when the load is high, there are more collisions in the network and the bandwidth of the network is not high enough. However, HIMAC makes multicast more reliable.

F. Number of Multicast Sessions

Figure 13 shows the impact of the number of multicast sessions. Every multicast session contains 1 sender and 10 receivers. For scenarios with 10 multicast sessions in the network, where every sender sends 200 packets per second, the traffic in the network is very heavy. 802.11 does not have any support for multicast or broadcast reliability. It always tries to broadcast packets as soon as the sender's channel is available regardless of the receivers' channels. However, HIMAC not only increases the transmission rate, but also enhances the delivery reliability. In HIMAC, the sender does not send packets until some receivers are ready to receive packets. Therefore, when the number of multicast sessions is high, the throughput of HIMAC is much higher than 802.11 although the latency of HIMAC slightly exceeds that of 802.11 for higher number of sessions. The throughput of HIMAC is 74% higher than the one of 802.11 in the scenarios with two multicast sessions.

G. Number of Receivers

Figure 14 shows the impact of number of receivers. We observe that HIMAC always outperform 802.11 in all cases. However, when the number of receivers is increased, the improvement of HIMAC reduces. The reason is that when the number of receivers increases, the size of the multicast tree becomes larger and the average number of receivers for a data packet increases in the MAC layer, which reduces the excepted rate for HIMAC and reduces the performance of HIMAC. Another reason is that when the number of the receivers increases for a sender in MAC layer, the probability that all the receivers have overheard the data packet reduces. So, the sender sends more packets when the number of receivers increases, which consumes more network resources and reduces the performance of HIMAC. When the number of the receivers is 5, the throughput of HIMAC is 74% higher than 802.11. The MAC latency of HIMAC is $40-53\%$ lower than 802.11.

H. CS/RX range ratio

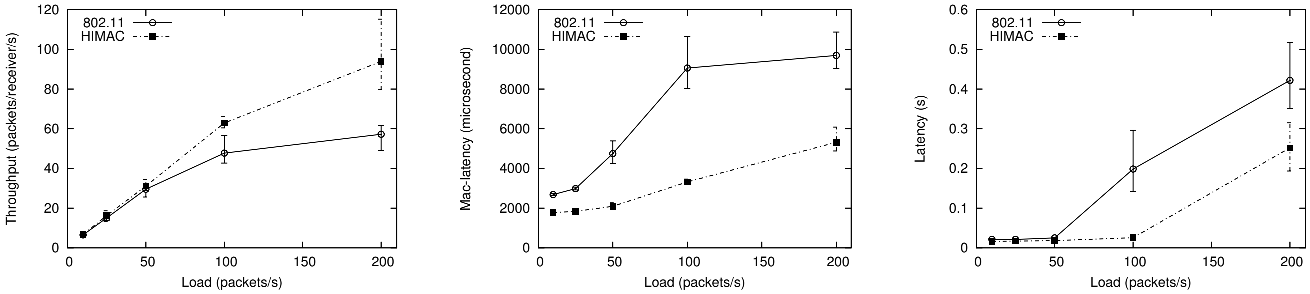
The ratio of carrier sensing threshold and the transmission/reception range depends on factors such as the hardware and the environment [19]. Figure 15 shows the impact of CS/RX range ratio. In this simulation, we fix receiving power threshold and change carrier sensing threshold to change CS/RX range ratio. When the CS/RX range increases, the throughput of HIMAC slightly increases while the throughput of 802.11 slightly reduces. The reason is that when the CS/RX range ratio increases, the interference (noise) reduces in networks because carrier sensing range becomes larger, which makes HIMAC more opportunity to use higher rates to transmit data packets and increase throughput. On the other hand, large CS range reduces the total number of simultaneous MAC layer multicast transmissions, which reduces the throughput of multicast. HIMAC outperforms 802.11 for all CS/RX range ratios. When the ratio is 2.25, the throughput of HIMAC is 64% higher than 802.11.

VI. Related Work

Prior research in wireless multicast and braodcast has focused on the transport layer [20], [21], network layer [16], [22]–[29] and MAC layer [5]–[11], [25], [30]–[33]. Time-varying channels and rate control has been studied by researchers for unicast transmissions [13]–[15].

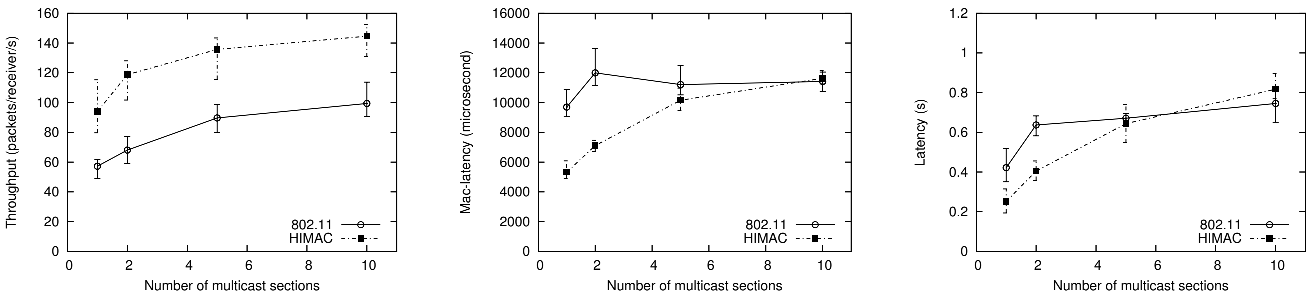
Transport Layer And Network Layer Multicast Protocols: End-to-end multicast and braodcast protocols [20], [21] address mechanisms to reliably recover lost packets and minimize overhead of information exchange among nodes. Network layer multicast protocols [16], [22]–[24] address efficiency and reliability considering various aspects of wireless links such as mobility and shared broadcast medium. Some multicast and braodcast routing protocols [25]–[27] address the issue of energy efficiency. Zhou and Singh [29] presented a new multicast model based on the content of the multicast data for ad hoc wireless networks. Nagy and Singh [28] investigated how to efficiently multicast data to mobile users in cellular networks. Bhatia and Li [34] analyzed techniques for maximizing multicast rate in multi-hop wireless networks. Alothough most transport and network layer multicast protocols work with any MAC layer protocols, the efficiency of the MAC layer protocol affects the efficiency of higher layer protocols.

MAC Layer Multicast Protocols: IEEE 802.11 MAC protocol implements multicast using broadcast. As the 802.11 broadcast is unreliable, several protocols [5]–[10], [31], [32] have been proposed to improve reliability. Kuri and Kasera [5] proposed a reliable multicast



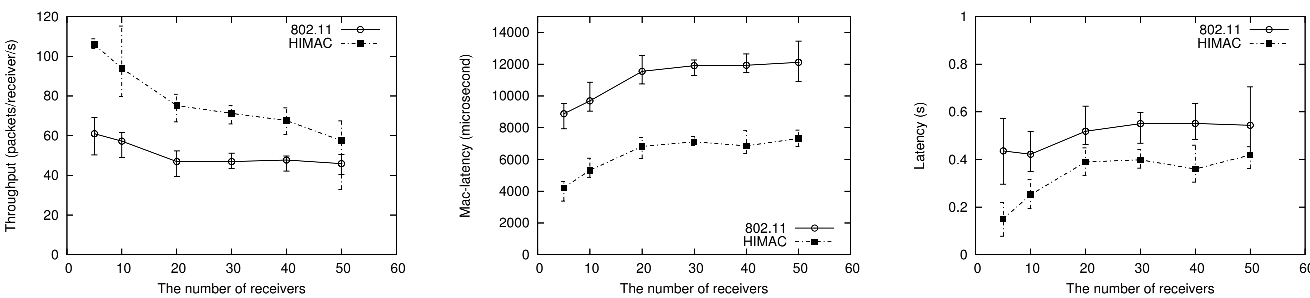
(a) Throughput (b) MAC Latency (c) Latency

Fig. 12. Network Load: With increasing network load, HIMAC's performance improvement becomes more significant.



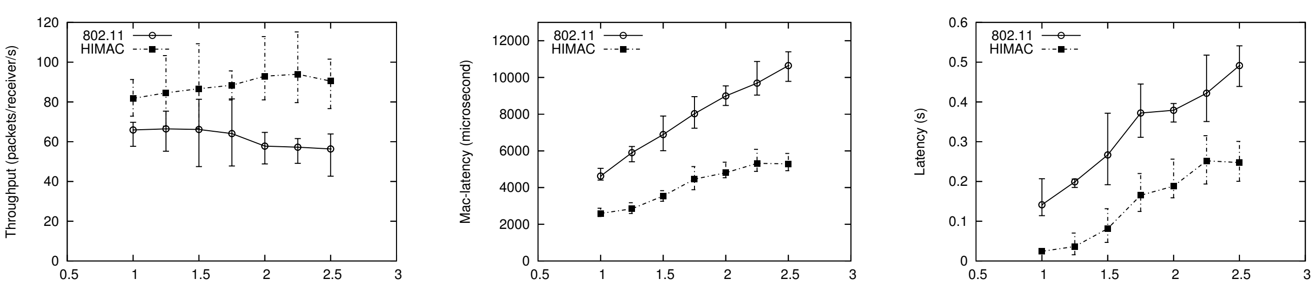
(a) Throughput (b) MAC Latency (c) Latency

Fig. 13. Number of Multicast Sessions: The throughput of HIMAC is 74% higher than the one of 802.11 in the scenarios of 2 multicast sessions.



(a) Throughput (b) MAC Latency (c) Latency

Fig. 14. Number of Receivers: When the number of the receivers is 5, the throughput of HIMAC is 74% higher than the one of 802.11 MAC



(a) Throughput (b) MAC Latency (c) Latency

Fig. 15. CS/RX Range Ratio: HIMAC outperforms 802.11 significantly for all CS/RX range ratios.

protocol for WLANs. This protocol is based on many assumptions, such as direct communication and perfect synchronization, which are not suitable for ad hoc networks. And it uses negative acknowledgments which can not confirm reliable transmission because when the receiver does not receive both RTS and the data packet, the receiver will not send NAK to the sender. Tang and Gerla [31], [32] extended the broadcast mechanism of 802.11 that tries to confirm that at least one receiver receives the broadcast packet in ad hoc networks. In [6], Tang and Gerla proposed BMW (Broadcast Medium Window) protocol which implements broadcast based on unicast and lets receivers overhear packets. BMW is a scalable protocol, but it can introduce arbitrary long latency for data packets. In [7], Sun *et al.* proposed BMMM (Batch Mode Multicast MAC) protocol to implement reliable MAC layer multicast. Basically, BMMM needs n pairs of RTS/CTS exchange and n pairs of RAK (Request for ACK)/ACK exchange for the transmission of one data packet to n receivers. This approach is not scalable and is not practical in high-traffic networks. Some MAC layer multicast/braodcast protocols, such as BPBT [8], RMAC [9], and 80211MX [10], use busy-tone to implement multicast reliability. Busy-tone can prevent data frame collisions and solve hidden terminal problem. However, Busy-tone requires a separate channel, which increases the hardware complexity.

Singh *et al.* [25] proposed a MAC layer protocol to support power-aware broadcasting in mobile ad hoc networks. Jaikaeo and Shen [30] investigated the benefits and impacts of using directional antennas for multicast communications in ad hoc networks. Chaporkar *et al.* [11], [33] proposed algorithms for maximizing throughput for MAC layer wireless multicast using busy tones. Their basic idea is that after the sender sends RTS to receivers on the message channel, all receivers that are ready to receive the data packet send a busy tone on a busy-tone channel. Then, the sender estimates the number of ready receivers by measuring the power of the busy tone signal. If the power of the busy tone signal is higher than the threshold, the sender will send data packets; otherwise, the sender retransmits RTS. The power threshold is decided by the queue length of the sender, which makes this protocol simple. The shorter the queue length, the higher the threshold to confirm more receivers are ready to receive the data packet. This protocol has three problems. First, it is difficult to accurately estimate the number of responding neighbors by measuring the power level because of fading in wireless links [12]. Second, it can introduce arbitrary long latency for data packets. For example, if just one packet is in the sender's queue, the threshold is very high. But if the link quality is bad and no other data packets come into the sender's queue, this packet will always stay in the sender's queue because the power level is always lower than the threshold. Third, if the network's traffic is heavy, this protocol will let senders set threshold as 0 and force senders to transmit multicast data packets as fast as possible, which can create a lot of collisions in the network and make the network condition worse and eventually reduce the throughput of multicast transmissions. Although the authors have shown their analysis for a single-hop network, we believe that the protocol has a divergent behavior in multi-hop scenarios. Their simulations are also limited to single-hop scenarios.

MAC layer Multirate Unicast Protocols: As the IEEE 802.11 physical layer supports multi-rate transmissions, several unicast protocols have been proposed to exploit this capability. In [13], Kamerman and Monteban present the ARF (Auto Rate Fallback) protocol for IEEE 802.11, used in Lucent's WaveLAN II devices. In ARF, senders increase transmission rate after consecutive transmission successes and reduce rate after consecutive transmission failures. RBAR (Receiver Based Auto Rate) protocol is proposed in [14]. The key idea of RBAR is to let the receiver measure the channel quality. The receiver then determines the transmission rate for the data packet as the highest feasible value allowed by the channel condition. Sadeghi *et al.* proposed OAR (Opportunistic Auto Rate) protocol in [15]. The major difference between OAR and RBAR is that OAR lets the sender send more packets when the channel quality is high.

VII. Conclusions

Time-varying channels and multiple physical layer data rates have never been considered in the design of MAC layer multicast protocols. Multicasting in 802.11 is achieved by physical layer broadcast which suffers from the following two problems identified in this paper: Channel-state Indifference and Demand Ignorance. The proposed HIMAC solution addresses these problems using the Unary Channel Feedback (UCF) and Unary Negative Feedback (UNF). HIMAC provides a novel approach to the design of an efficient, scalable and robust MAC layer. We analyze the solution using measurements to support our claims. Extensive performance evaluation with realistic Rayleigh fading model in *ns-2* simulations show that HIMAC performs significantly better than 802.11 in terms of throughput, MAC latency, and end-to-end latency. As part of our future work, we are studying the performance of unicast as a special case of our approach. We are also working on emulating HIMAC in large scale testbeds (ORBIT at Rutgers) by obtaining the receivers' feedback over Ethernet, as current NICs do not allow changes to the MAC layer. Of the two problems identified in this paper, the Demand Ignorance problem is difficult to solve in its full generality and is left as open research problems.

In the future, we will try to solve the problem of hidden terminal nodes who can only hear the UCF or UNF generated by the multicast receivers. In HIMAC, these nodes do not need to be silent because it is impossible to set NAV in UCF or UNF.

References

[1] Y. Zhu and T. Kunz, "MAODV Implementation for NS-2.26," in *Technical Report SCE-04-01*, Department of Systems and Computer Engineering, Carleton University, Canada, January 2004.

[2] D. S. Kim and D. Du, "Multirate Multicast Switching Networks," *Theor. Comput. Sci.*, vol. 261, no. 2, pp. 241–251, 2001.

[3] G. I. Kwon and J. W. Byers, "Smooth Multirate Multicast Congestion Control," in *INFOCOM*, 2003.

[4] S. Sarkar and L. Tassiulas, "Fair Distributed Congestion Control in Multirate Multicast Networks," *IEEE/ACM Trans. Netw.*, vol. 13, no. 1, pp. 121–133, 2005.

[5] J. Kuri and S. K. Kasera, "Reliable Multicast in Multi-Access Wireless LANs," in *INFOCOM*, 1999, pp. 760–767.

[6] K. Tang and M. Gerla, "MAC Reliable Broadcast in Ad Hoc Networks," in *Proceedings of IEEE MILCOM*, 2001, pp. 1008–1013.

[7] M. T. Sun, L. Huang, A. Arora, and T. H. Lai, "MAC Layer Multicast in IEEE 802.11 Wireless Networks," in *ICPP*, 2002.

[8] C. Y. Chiu, E. H. Wu, and G. H. Chen, "A Reliable and Efficient MAC Layer Broadcast (Multicast) Protocol for Mobile Ad Hoc Networks," in *Global Telecommunications Conference*, 2004, pp. 2802–2807.

[9] W. Si and C. Li, "RMAC: A Reliable Multicast MAC Protocol for Wireless Ad Hoc Networks," in *ICPP'04*, 2004, pp. 494–501.

[10] S. Gupta, V. Shankar, and S. Lalwani, "Reliable Multicast MAC Protocol for Wireless LANs," in *Proc. IEEE ICC03*, May 2003, pp. 93–97.

[11] P. Chaporkar, A. Bhat, and S. Sarkar, "An Adaptive Strategy for Maximizing Throughput in MAC Layer Wireless Multicast," in *MobiHoc'04*, Roppongi, Japan, May 2004, pp. 256–267.

[12] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Prentice Hall, 1996.

[13] A. Kamerman and L. Monteban, "WaveLAN II: A High-Performance Wireless LAN for the Unlicensed Band," *Bell Labs Technical Journal*, pp. 118–133, Summer 1997.

[14] G. Holland, N. Vaidya, and P. Bahl, "A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks," in *ACM MOBICOM*, 2001.

[15] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. W. Knightly, "OAR: An Opportunistic Auto-Rate Media Access Protocol for Ad Hoc Networks," *Wireless Networks*, vol. 11, no. 1-2, pp. 39–53, 2005.

[16] E. M. Royer and C. E. Perkins, "Multicast Operation of the Ad Hoc On-Demand Distance Vector Routing Protocol," in *ACM MOBICOM*, Seattle, WA, Aug 1999, pp. 207–218.

[17] J. J. Garcia-Luna-Aceves and E. Madruga, "The Core-Assisted Mesh Protocol ," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1380–1394, Aug. 1999.

[18] S. Glisic, T. Poutanen, W. Wu, G. Petrovic, and Z. Stefanovic, "New PN Code Acquisition Scheme for CDMA Networks with Low Signal-to-Noise Ratios," *IEEE Transactions on Communications*, vol. 47, no. 2, pp. 300–310, Feb 1999.

[19] G. Zhou, T. He, J. A. Stankovic, and T. F. Abdelzaher, "RID: Radio Interference Detection in Wireless Sensor Networks," in *INFOCOM*, March 2005.

[20] R. Chandra, V. Ramasubramaniam, and P. Birman, "Anonymous Gossip: Improving Multicast Reliability in Mobile Ad-Hoc Networks," in *ICDCS-21*, Phoenix (Mesa), Arizona, Apr 2001.

[21] E. Pagani and G. Rossi, "Reliable Broadcast in Mobile Multihop Packet Networks," in *MOBICOM*, 1997, pp. 34–42.

[22] C. C. Chiang, M. Gerla, and L. Zhang, "Forwarding Group Multicast Protocol (FGMP) for Multihop, Mobile Wireless Networks," *Baltzer Cluster Computing*, vol. 1, no. 2, pp. 187–196, 1998.

[23] S. J. Lee, M. Gerla, and C. C. Chiang, "On-demand Multicast Routing Protocol," in *WCNC*, New Orleans, Sep 1999, pp. 1298–1304.

[24] T. Ozaki, J. B. Kim, and T. Suda, "Bandwidth-Efficient Multicast Routing for Multihop, Ad-Hoc Wireless Networks," in *INFOCOM*, 2001, pp. 1182–1191.

[25] S. Singh, C. S. Raghavendra, and J. Stepanek, "Power-Aware Broadcasting in Mobile Ad Hoc Networks," in *PIMRC*, 1999.

[26] P.-J. Wan, G. Calinescu, X. Li, and O. Frieder, "Minimum-Energy Broadcast Routing in Static Ad Hoc Wireless Networks," in *INFOCOM*, 2001, pp. 1162–1171.

[27] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks," in *INFOCOM*, 2000, pp. 585–594.

[28] M. Nagy and S. Singh, "Multicast Scheduling Algorithms in Mobile Networks," *Cluster Computing*, vol. 1, no. 2, pp. 177–185, 1998.

[29] H. Zhou and S. Singh, "Content Based Multicast (CBM) in Ad Hoc Networks," in *MobiHoc*, Boston, 2000, pp. 51–60.

[30] C. Jaikaeo and C. Shen, "Multicast Communication in Ad Hoc Networks with Directional Antennas," in *ICCCN 2003*, Dallas, Texas, 2003.

[31] K. Tang and M. Gerla, "MAC Layer Broadcast Support in 802.11 Wireless Networks," in *Proceedings of IEEE MILCOM*, 2000.

[32] ——, "Random Access MAC for Efficient Broadcast Support in Ad Hoc Networks," in *WCNC*, 2000.

[33] P. Chaporkar and S. Sarkar, "Stochastic Control Techniques for Throughput Optimal Wireless Multicast," in *Control and Decision Conference (CDC)*, Maui, Hawaii, Dec 2003.

[34] R. Bhatia and L. Li, "Characterizing Achievable Multicast Rates in Multi-Hop Wireless Networks," in *MobiHoc'05*, 2005, pp. 133–144.