Efficient Multicast Supporting in Multi-rate Wireless Local Area Networks *

Yongho Seok and Yanghee Choi

School of Computer Science and Engineering, Seoul National University
{yhseok, yhchoi}@mmlab.snu.ac.kr

Abstract. Wireless local area networks(LAN) have become popular due to their capability of supporting the high data rates required in ubiquitous computing. These high data rates are possible through the use of new modulation schemes that are optimized for different channel conditions. In particular, IEEE 802.11b supports a variety of different data rates, i.e. 11Mbps, 5.5Mbps, 2Mbps and 1Mbps, depending on the channel conditions between the access point(AP) and the mobile station. All mobile stations share the same wireless channel or frequency, so the bottleneck point of wireless LAN system is the traffic load of the lowest data rate connections. Since wireless LAN supports higher data rates, it is more attractive for ubiquitous multimedia services. Multimedia services generally require high bandwidth and most of them utilize the multicast system. Since most multimedia services are provided by real-time communications, reduced end-to-end packet delay and delay jitter are more important than reliable packet transmission. However the data rate of the multicast communication used in most commercial products is generally set to be low by the administrator, who only considers the coverage of the AP, and its value is generally fixed. In this environment, data rate which is selected for the transmission of multicast packets is a very important factor in determining the system performance of the wireless LAN, since multimedia services require a large bandwidth. Thus, a new algorithm is necessary to dynamically select the data rate of the multicast according to the traffic load of the wireless LAN. In this paper, we present two naive data rate selection algorithms which allow the data rate to be dynamically adjusted, and a more novel data rate selection algorithm based on the two primitive algorithms. We show that our suggested algorithm achieves high performance in wireless LAN and efficiently supports high quality multimedia services, through the use of the $NS-2$ simulation in a wireless LAN environment based on IEEE 802.11b.

1 Introduction

The IEEE 802.11a and IEEE 802.11b media access control(MAC) protocols provide a physical-layer multi-rate capability [1]. The original IEEE 802.11 protocol supports a single base rate, typically 2Mbps. With the multi-rate enhancement, the data transmission can take place at various rates according to the channel

* This work was supported in part by the Brain Korea 21 project of Ministry of Education and in part by the National Research Laboratory project of Ministry of Science and Technology, 2002, Korea.

conditions. Higher data rates than the base rate are possible when the signal-to-noise ratio(SNR) is sufficiently high. With IEEE 802.11a the set of possible data rates includes 6, 9, 12, 18, 24, 36, 48 and 54 Mbps whereas for IEEE 802.11b the set of possible data rates includes 1, 2, 5.5 and 11 Mbps. As the multi-rate enhancements are physical layer protocols, MAC mechanisms are required to exploit this capability. The auto rate fallback(ARF) protocol is the commercial implementation of a MAC that utilizes this feature [2]. With ARF, senders attempt to use higher transmission rates after consecutive transmission successes and revert to lower rates after failures. Under most channel conditions, this multi-rate enhancement using ARF provides a performance gain over the pure IEEE 802.11 single base rate.

Since the ARF protocol selects the data rate according to the channel conditions between the AP and mobile station, it can only be used in point-to-point communications, such as in the case of unicast. In the case of the point-to-multipoint communications(e.g., multicast and broadcast), it is difficult to determine the data rate, because the link characteristics between the AP and each mobile station can vary. In current commercial products, the administrator selects the data rate of the point-to-multipoint connection, which is used to provide network connectivity to all mobile stations covered by the AP. Because the coverage of the AP is inversely proportional to the transmission data rate, the administrator selects the proper data rate according to the distance between the different APs. As the distance between the APs is increased, the data rate of the point-to-multipoint connection has to be reduced in order to compensate for the increased range that the AP has to cover. This simple approach dose not efficiently support point-to-multipoint connections, due to the characteristics of wireless LAN explained below.

All mobile stations in the basic service set(BSS) share the same wireless channel or frequency, by using the Carrier Sense Multiple Access-Collision Avoidance MAC protocol. In this environment, the bottleneck point of the wireless LAN system is the traffic load of the lowest data rate connections, which may involve either point-to-point connections or point-to-multipoint connections. When the number of packets transmitted to mobile stations using 1Mbps data-rate is increased, the resulting deterioration in throughput, delay and packet loss will always be greater than that in the case of an 11Mbps data-rate, even if the total amount of transmitted traffic is the same in both cases. Thus, to maximize throughput, we need to consider the performance degradation of wireless LAN resulting from the traffic load of the lowest data rate connections between the AP and mobile stations. In the case of point-to-point communications, the appropriate data rate can be set individually for each mobile station, after taking the current channel conditions, such as the SNR, into consideration. However, in the case of point-to-multipoint communication, the data rate is almost always set to a fixed, low value by the administrator, who only takes the coverage area of the AP into consideration.

We propose an algorithm which can be used to dynamically select data rate of the multicast, which determines the overall performance of the wireless LAN system and the quality of service(QoS) of the multimedia application. The remainder of this paper is organized as follows. We start in Section 2 by providing some background on some data rate selection algorithms used for unicast packet transmission in multi-rate wireless LAN. The proposed data-rate selection algorithms for multicast packet transmission are described in Section 3. Section 4 presents the results of the simulation experiments. Finally, we summarize and conclude this paper in Section 5.

2 Related Work

Most commercial implementations that exploit the multi-rate capability of IEEE 802.11b and IEEE 802.11a are termed ARF [2]. The ARF scheme proposed in [2] is based on keeping track of the timing function and missed acknowledgment(ACK) messages. Operation at the maximum data rate is considered to be the default. When an ACK is missed for the first time following earlier successful transmissions, the first retry transmission is still performed at the same rate. When the ACK is missed again, the second retry and subsequent transmissions are performed at the fallback rate, which is half of the previous date rate. In the [3], a rate adaptive MAC protocol, called the receiver-based auto rate(RBAR) protocol, is presented. The novelty of RBAR is that its rate adaptation mechanism is situated in the receiver instead of in the sender. This is in contrast to the ARF scheme in which the sender decides the data rate. RBAR is based on the RTS/CTS mechanism and it is better because it results in a more efficient channel quality estimation which is then reflected in a higher overall throughput.

3 Data-rate Selection Algorithm of Multicast Packet Transmission

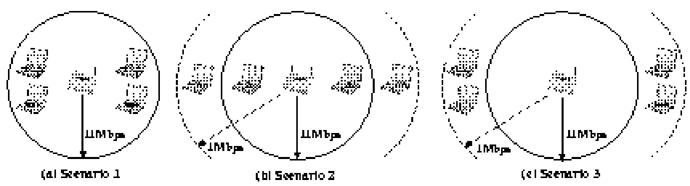

Fig. 1. Multicast issues in multi-rate wireless LAN

The main issue in transmitting multicast packets is selecting the appropriate data rate. Currently, in most commercial APs that support the IEEE 802.11b protocol, the administrator selects a fixed data rate. The drawback of this approach is that selecting a fixed data rate is not an effective solution. In particular, it is very inefficient to set the data rate to a certain low value so that all mobile stations can receive the multicast packets from the AP. In most multicast applications, perfect reliability is not needed. In addition, such reliability is not guaranteed even with a low data rate, due to the inherent limitations associated with wireless communication. Since multicast applications generally require high bandwidth, transmitting at a low data rate leads to a loss of throughput. In Figure 1 (a), if the data rate of the multicast is set to 11Mbps, the four mobile stations in the multicast group can receive the multicast packets. However, in Figure 1 (c), the four mobile stations have difficulty in receiving the multicast packets, since they are beyond the coverage of the 11Mbps data rate. In Figure 1 (b), two of these mobile stations are located at a short enough distance from the AP to receive the data, while the other two mobile stations are not able to receive the data. If the data rate is reduced to 1Mbps so that all four stations can receive data, the wireless LAN suffers from throughput degradation problems resulting from the low data rate.

3.1 Fixed Data-rate

Most currently used wireless AP devices supporting IEEE 802.11b, transmit packets at a fixed data rate. The data rate is determined by the AP administrator according to the distance between the APs. Wider coverage for an AP is achieved by lowering the data rate, if the distance between the APs is great. The shortcoming of this simple approach is to mistakenly use a low data rate, whereas a high data rate could be used even when the all of the mobile stations in the group are located close to the AP. In order to solve this problem, a technique that applies a different data rate to each different multicast group is considered in next subsection.

3.2 Maximum Data-rate Covering the All Mobile Stations of Group

In this subsection, we consider a technique that maintains the lists of mobile stations for each multicast group. The transmission of multicast packets is carried out at the lowest data rate among data rates of all mobile stations in the list of the multicast group. This technique partially solves the problem resulting from the use of a fixed data rate, however, it also has the disadvantage of having to maintain a list of mobile stations in each group and the lowest data rate among them. Furthermore, the lowest data rate will always be selected, in order to be able to transmit data to all of the mobile stations in the group, even if most multicast applications require higher bandwidth. For example, let us consider the case of a real-time multicast requiring a 1.5Mbps bandwidth. If one of the mobile stations in the multicast group has a 1Mbps link and the others all have 11Mbps links, then the problem of which data rate to select becomes apparent. The 1Mbps data rate would enable every mobile station to receive the data, however selecting this option would result in low throughput due to high resource utilization($\frac{Traffic}{Capacity} \simeq \frac{1.5Mbps}{1Mbps}$). To improve the throughput, selecting the 11Mbps data rate would be better, although some mobile stations would have difficulty in receiving the data. However, a mobile station situated far from the AP is not entirely prevented from receiving the data, even at the 11Mbps data rate. The only problem is that the bit error rate will be high. Consequently, we need to provide flexibility in the selection of the data rate, since most multicast applications do not require high reliability.

3.3 Adaptive Data-rate based on the Average Transmission Time of Packets

We propose an algorithm which provides a means of selecting the data rate of a multicast packet according to the traffic load. If setting the data rate such that all mobile stations in a group can receive data results in a degradation of the wireless LAN's performance, we prefer to increase the data rate in order to improve the overall performance of the wireless LAN. However, in this case, some mobile stations in a group may not receive data packets, since the IEEE 802.11 MAC protocol provides reliability using ACK packets in the case of unicast packets, but not in the case of multicast or broadcast packets [5]. This is not a serious drawback since, in multicast applications, reducing delay and delay-jitter is more important than providing reliability, and at application level, UDP is used, rather than TCP, for this reason.

For our proposed technique, we used a new algorithm for maintaining queues, instead of the existing *DropTail* buffer management algorithm. We choose the packet with the longest transmission time for dropping when the queue is full. Hence, by this algrorithm, we can maximize the number of available entries and decrease the probability of dropping packets accordingly. When the network layer has a packet to transmit, it calls the enqueue function of our queue management system. In the enqueue function, if the queue is full, we choose the packet which has the longest transmission time based on the channel state monitor information, such as the data-rate of each mobile station and then drop it. When the MAC protocol wants to transmit a packet, it calls the dequeue function of our queue management system. In this case, the dequeue operation of the unicast packet is simply to transmit the packet with current data rate between the AP and the packet destination. However, the dequeue operation of a multicast packet requires an additional mechanism in order to decide the appropriate data rate. The data rate decision mechanism will be shown in Algorithm 2.

Table 1. Notations for algorithm description

Notation	Description
$Q.Length()$	return the queue length (*packets*)
$Q.Insert(P)$	insert the *packet* P into the queue
$Q.Lookup(i)$	return the *i'th packet* in the queue
$Q.Remove(P)$	delete the *packet* P in the queue
$P.Size()$	return the size of *packet* P
$P.NextHop()$	return the next hop address of *packet* P
$Bit\_Rate(k)$	the host k's bit rate

In Algorithm 1, we present the proposed queue management algorithm using the notation shown in Table 1. Algorithm 1 operates when a new packet (P) arrives at its queue (Q). The queue management module inserts the arriving packet (P) into the queue (Q). If the queue is full, we determine which packet to be dropped according to the following rules. The overflow of the queue implies that the number of incoming data packets for the wireless system is greater than the number of outgoing packets. Therefore, we choose the packet which would have the longest transmission time to be dropped, so that we can reduce the packet drop probability. Since the upper bound of the queue length commonly uses the number of packets instead of the number of bytes, reducing the average packet transmission time results in an increase in the available queue space. Lines 5-13 in Algorithm 1 shows how to look for the packet which has the longest transmission time ($P.Size()/Bit\_Rate(k)$) in the queue Q. For multicast packets, the data rate is set to the highest value among the data rates of all the mobile stations in the group. In the case of unicast, there is only one receiver, but in the case of multicast there are several. When comparing the dropping priority between a unicast packet and a multicast packet, the transmission time of the multicast packet is based on the highest data rate, so that just one receiver is able to receive the packet. Since the multicast connection is considered to be a point-to-point connection, as in the case of unicast, this mechanism can provide fairness of throughput between unicast and multicast. At Line 14, the average transmission time of dropped packets is evaluated using the exponentially weighted moving average(EWMA). Namely, the most recently dropped packet has the weight of w and the cumulated average has $1 - w$.

Algorithm 1 *Enqueue*(P) P : newly arriving packet

```
1:  Q.Insert(P);
2:  if Q.Length() ≥ QueueLimit then
3:    i ← 1;
4:    Max_TxTime ← 0;
5:    while i ≤ Q.Length() do
6:      P_tmp ← Q.Lookup(i);
7:      TxTime ←
            P_tmp.Size()/Bit_Rate( P_tmp.NextHop() );
8:      if TxTime > Max_TxTime then
9:        Max_TxTime = TxTime;
10:       P_victim ← P_tmp;
11:     end if
12:     i ← i + 1;
13:   end while
14:   Avg_TxLimit ← w · Max_TxTime + (1 − w) · Avg_TxLimit;
15:   Q.Remove(P_victim);
16:   Drop(P_victim);
17: end if
```

The transmission data rate of a multicast packet is determined by this average transmission time as in the Algorithm 2. *Avg_TxLimit* presents the upper bound of transmission time that packets in the queue must not exceed. Once the packet (P_{victim}) to be dropped has been chosen, we remove this packet from the queue (Q). This priority queue management algorithm, based on the transmission time of the packet, gives the lower priority to mobile stations with lower-data rates, when the queue is full. However, small sized packets, such as those used in voice or interactive applications, get high priority for transmission, even though their data rate is low, since little time is required for transmitting small-sized packets, even at a low data rate.

In Algorithm 2, the transmission data rate of a multicast packet is determined as follows. As in the case of the queue management algorithm described in Algorithm 1, which drops the packet with the longest transmission time when the queue is full, a similar method needs to be applied here. The transmission time of a multicast packet should be shorter than the average transmission time of all dropped packets (*Avg_TxLimit*), in order to agree to giving higher priority of entering the queue to the packet with shorter transmission time. In this way, resources are allocated fairly to both multicast and unicast packets. Thus, when the queue length exceeds a certain threshold(*Threshold*), the data rate of a multicast packet is set, such that the transmission time of the multicast packet is less than the average transmission time of all dropped packets(Lines 3-11 in Algorithm 2). If there are many such data rates, the lowest one is selected so that the largest number of mobile stations are served.

Algorithm 2 $Data-rate\ Selection\ of\ Multicast\ Packet\ P$

```
1: Multicast_rate ← 1Mbps;
2: if Q.Length() ≥ Threshold then
3:   if P.Size()/1Mbps ≤ Avg_TxLimit then
4:     Multicast_rate ← 1Mbps;
5:   else if P.Size()/2Mbps ≤ Avg_TxLimit then
6:     Multicast_rate ← 2Mbps;
7:   else if P.Size()/5.5Mbps ≤ Avg_TxLimit then
8:     Multicast_rate ← 5.5Mbps;
9:   else if P.Size()/11Mbps ≤ Avg_TxLimit then
10:    Multicast_rate ← 11Mbps;
11:  end if
12: end if
```

4 Simulation Experiments

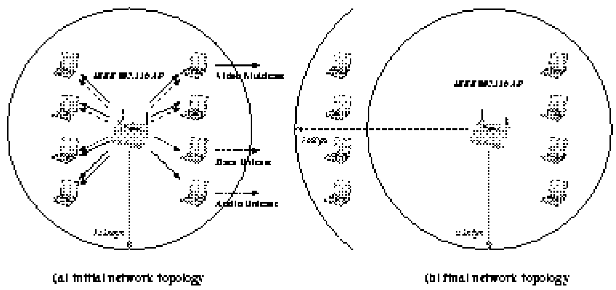

Fig. 2. Simulation scenario

The simulation results in this paper were generated using the $ns-2$ network simulator, with extensions from NOAH, a wireless routing protocol that supports direct communication between APs and mobile stations. In this simulation, the available data rates are based on IEEE 802.11b and are set to 1 Mbps, 2 Mbps, 5.5 Mbps, and 11 Mbps. The values for the data rates were chosen based on the distance ranges specified in the *Orinoco*^{TM} 802.11b card data sheet [6]. Figure 2 (a) represents the initial network topology. It is composed of one access point(AP) and 8 mobile stations. In Figure 2 (a), all mobile stations are close enough to the AP so that they can all be served by the AP using the 11 Mbps data rate. When the simulation starts, half of the 8 mobile stations begin to move away from the AP with a velocity of 0.5 m/s. Consequently, their data rates are successively reduced firstly from 11 Mbps to 5.5 Mbps, then to 2 Mbps and finally down to 1 Mbps. Figure 2 (b) shows 4 the situation in which the 4 mobile stations move, while the other 4 mobiles stations stay in place, and these stations are served using the 1 Mbps and 11 Mbps data rates, respectively.

We performed our simulation using 3 different applications(Video, Audio, and Data traffic) to show the various internet traffic characteristics. In this simulation, the video application uses multicast routing while the other applications use unicast routing. The mobile stations joining the multicast group for the video application comprise 4 moving mobile stations and 2 stationary mobile stations. Each of the data and audio applications has 4 connections, 2 of them involve moving mobile stations, and the other 2 involve stationary mobile stations. The video traffic is CBR generating 1.5 Mbps with a packet size of 1400 bytes. The audio traffic is exponential On/Off, which can be configured to behave as a Possion process, generating 256 Kbps with a packet size of 128 bytes. The data traffic is Pareto On/Off generating 512 Kbps with a packet size of 512 bytes. All traffic sources in the wired network are 2 hops away from the AP.

4.1 Simulation Result

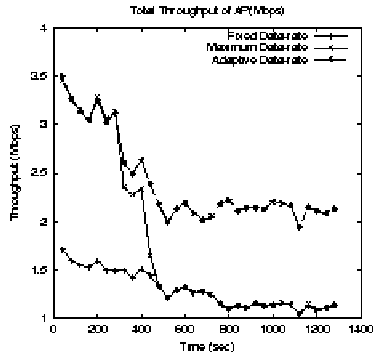

Fig.3. Total throughput of AP

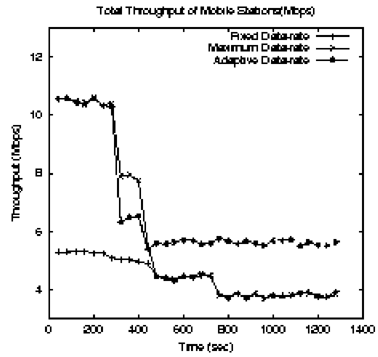

Fig.4. Total throughput of mobile stations

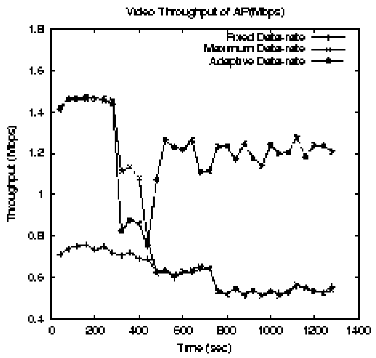

Fig.5. Video throughput of AP

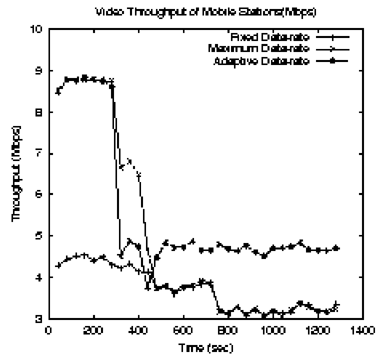

Fig.6. Video throughput of mobile stations

Figure 3 shows the total throughput of the AP, i.e. the total number of data bytes including Video, Audio and data traffic transmitted from the AP to mobile stations per second. The x-axis represents time ranging from 0 to 1400 seconds and the y-axis represents the throughput(Mbps). In this graph, we show 3 data rate selection algorithms for multicast packets, *Fixed Data-rate* (Section 3.1) which implements the multicast packet transmission with the 1Mbps fixed data-rate , *Maximum Data-rate* (Section 3.2) which implements the multicast packet transmission with the maximum data-rate covering all mobile stations in each group and *Adaptive Data-rate* (Section 3.3) which implements the multicast packet transmission with the adaptive data-rate based on the average transmission time of dropped packets.

First, when *Fixed Data-rate* is used, throughput hardly exceeds 1.5Mbps. This is due to the packet's fixed data rate of 1Mbps. Using *Maximum Data-rate* which maintains a list of the multicast packets' data rates for each multicast group, a relatively high throughput is obtained, particularly in the case where all of the mobile stations in a group are close to the AP. However, as the distance from the AP to the mobile stations increases, the packets' data rate decreases and, after 500(sec), it shows a similar throughput to that of *Fixed Data-rate*. Consequently, maintaining separate data rates per multicast group is not the most effective alternative. *Adaptive Data-rate* shows a throughput of over 2Mbps, even after 500s. It is shown that selecting the appropriate data rate for multicast, according to the traffic load of the wireless LAN, can lead to an increase in the throughput. In the proposed *Adaptive Data-rate* algorithms, the number of dropped packets can be reduced by using the packet transmission time for queue management, since in this way it is possible to reduce the length of the queue rapidly. Moreover, it is possible to dynamically adjust to the traffic conditions of the wireless LAN, by setting the data rate of the multicast packets after considering the average transmission time of the dropped packets.

The graph in Figure 4 shows the throughput in terms of the total amount data received by all mobile stations. The y-axis shows the total quantity of packets successfully received by all mobile stations for each packet transmitted by the AP. The graph shows similar results to those in Figure 3. If the data rate of the multicast packets is fixed (*Fixed Data-rate*), the receiving throughput of all mobile stations is maintained at 4 to 5Mbps. On the other hand, it suffers penalty in throughput by over 2Mbps after 500s compared with *Adaptive Data-rate*. Comparing *Maximum Data-rate* with *Adaptive Data-rate*, their throughput up until 500s is similar, as shown in the graphs in Figure 3. This result shows that determining the multicast data rate by means of the data rates of the mobile stations in the multicast group is quite effective, as long as the mobile stations are located close to the AP. However, when using this approach, no more gain in throughput was obtained after 500s when at least one terminal became so far from the AP that the data rate was set to a lower value in order to make all mobile stations receive data from the AP. In this case, the data rate has to be set to a completely new value, as in *Adaptive Data-rate*. Thus, when the overall performance of the wireless LAN system is degraded, the data rate should be increased to a higher value, even though some stations may not be able to receive the data.

Figure 5 shows the throughput of the video application transmitted from the AP to the mobile stations in the multicast group. In this graph, the *Adaptive Data-rate* algorithm achieves a higher video throughput than the other algorithm especially after 500s. This indicates that the client application in the mobile station is able to display a greater number of frames per second, with the result that, a greater number of video packets is transmitted from the AP by using

Adaptive Data-rate algorithm The displayed frames per second is a very important factor related to the QoS. Thus, when the other algorithms (*Fixed Data-rate* or *Maximum Data-rate* algorithm) are used instead of *Adaptive Data-rate*, the QoS of the video application, as well as the overall throughput of the wireless system, is degraded.

Figure 6 shows the throughput of the video application received by the mobile stations and is similar to the Figure 4. It is clear that, after 500s, the throughput obtained by the use of *Adaptive Data-rate* is the highest. Even though some mobile stations may not receive certain packets when *Adaptive Data-rate* is used, the total throughput for all mobile stations is not degraded. Furthermore, the throughput of the receiving mobile stations, as well as the throughput of the AP, is improved.

5 Conclusion

In this study, we proposed a new data rate selection algorithm for multicast packets (*Adaptive Data-rate*), which can provide high system performance in a multi-rate wireless LAN environment. In addition to improve its performance, this algorithm also provides the ubiquitous multimedia service with good service quality, through the efficient use of the high bandwidth available in a multi-rate Wireless LAN environment. The proposed data rate selection algorithm is able to dynamically adjust the data rate of the multicast packet transmission according to the traffic load of the wireless LAN. It consists of two primitive algorithms, a priority queue management algorithm based on packet transmission time, and a data rate selection algorithm for multicast packets based on the average transmission time of all dropped packets. Since wireless LAN supports high data rates, such as the 54Mbps rate defined in IEEE 802.11a, as multimedia traffic using the multicast system increases, the algorithms proposed in this study will become more important. We performed a simulation of the proposed algorithm using the $ns-2$ simulator, and were able to confirm the predicted results. The proposed algorithm contributes to the efficient utilization of resources, as well as increasing the throughput of the wireless LAN and improving the QoS of the multimedia service, by reducing the end-to-end packet delay and increasing the number of frames per second which can be viewed. How to guarantee the fairness in the proposed priority queue management algorithm is one of our future work.

References

1. IEEE Computer Society. "802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications". June 1997.
2. A. Kamerman and L. Monteban. "WaveLAN II: A high-performance wireless LAN for unlicensed band". Bell Labs Technical Journal, pages 118-133, Summer 1997.
3. G. Holland, N. Vaidya and P. Bahl. "A rate-adaptive MAC protocol for multi-hop wireless networks". ACM MOBICOM'01, July 2001.
4. B. Sadeghi, V. Kanodia, A. Sabharwal and E. Knightly. "Opportunistic Media Access for Multirate Ad Hoc Networks". ACM MOBICOM'02, September 2002.
5. J. Kuri and S. Kasera. "Reliable Multicast in Multi-access Wireless LANs". ACM/Kluwer Wireless Networks Journal, July/August 2001
6. Datasheet for ORiNOCO 11 Mbit/s Network Interface Cards, 2001. ftp://ftp.orinocowireless.com/pub/docs/ORINOCO/.