A Multiplex-Multicast Scheme that Improves System Capacity of Voice-over-IP on Wireless LAN by 100% *

Wei Wang, Soung C. Liew, Qixiang Pang
Department of Information Engineering
The Chinese University of Hong Kong
{wwang2, soung, qxpang }@ie.cuhk.edu.hk

Victor O. K. Li
Department of Electrical and Electronic Engineering
The University of Hong Kong
vli@eee.hku.hk

Abstract

Voice-over-IP (VoIP) is an important application on the Internet. With the emergence of WLAN technology and its various advantages compared with the traditional wired LAN, it is fast becoming the "last-mile" of choice for the overall Internet infrastructure. This paper considers the support of VoIP over 802.11b WLAN. We show that although the raw WLAN capacity can potentially support more than 500 VoIP sessions, various overheads bring this down to only 12 VoIP sessions when using GSM 6.10 codec. We propose a novel multiplexing scheme for VoIP which exploits multicasting over WLAN for the downlink VoIP traffic. This scheme can achieve nearly 100% improvement in system capacity. In addition, we present results showing that the delay and delay jitter introduced by the proposed scheme are small. We believe that the scheme can reduce the blocking probability of VoIP sessions in an enterprise WLAN significantly.

Keywords – VoIP; Wireless LAN; 802.11; Access Point; Capacity; Multicasting; Multiplexing; Internet Telephone; Wireless Communications

1. Introduction

Voice over IP (VoIP) is one of the fastest growing Internet applications today [1]. It has two fundamental benefits compared with voice over traditional telephone networks. First, by exploiting advanced voice compression techniques and bandwidth sharing in packet-switched networks, VoIP can dramatically improve bandwidth efficiency. Second, it facilitates the creation of new services that combine voice communication with other media and data applications like video, white boarding and file sharing.

At the same time, driven by huge demands for portable access, the wireless LAN (WLAN) market is taking off quickly. Due to its convenience, mobility, and high-speed access, WLAN represents an important future trend for "last-mile" Internet access.

The most popular WLAN standard currently is IEEE 802.11b, which can support data rates up to 11Mbps. A VoIP stream typically requires less than 10Kbps. Ideally, the number of simultaneous VoIP streams that can be supported by an 802.11b WLAN is around 11M/10K = 1100, which corresponds to about 550 VoIP sessions, each with two VoIP streams. However, it turns out that the current WLAN can only support no more than a few VoIP sessions. For example, if GSM 6.10 codec is used, the maximum number of VoIP sessions that can be supported is 12, a far cry from the estimate. This result is mainly due to the added packet-header overheads as the short VoIP packets traverse the various layers of the standard protocol stack, as well as the inefficiency inherent in the WLAN MAC protocol, as explained below.

A typical VoIP packet at the IP layer consists of 40-byte IP/UDP/RTP headers and a payload ranging from 10 to 30 bytes, depending on the codec used. So the efficiency at the IP layer for VoIP is already less than 50%. At the 802.11 MAC/PHY layers, the drop of efficiency is much worse. Consider a VoIP packet with 30-byte payload. The transmission time for it at 11 Mbps is $30 * 8 / 11 = 22\ \mu\mathrm{sec}$. The transmission time for the 40-byte IP/UDP/RTP header is $40 * 8 / 11 = 29\ \mu\mathrm{sec}$. However, the 802.11 MAC/PHY layers have additional overhead of more than 800 $\mu\mathrm{sec}$, attributed to the physical preamble, MAC header, MAC backoff time, MAC acknowledgement, and inter-transmission times of packets and acknowledgements. As a result, the overall efficiency drops to less than 3%.

This paper proposes a voice multiplexing scheme to overcome the large overhead effect of VoIP in WLAN. Our scheme makes use of the features of the multicast mode of WLAN. We will show that the number of VoIP sessions that can be supported can be doubled with this simple technique, while maintaining small delay.

* This work is sponsored by the Areas of Excellence scheme established under the University Grant Committee of the Hong Kong Special Administrative Region, China (Project Number AoE/E-01/99).

2. VoIP Multiplex-Multicast Scheme

2.1 System Architecture

An 802.11 WLAN is referred to as the basic service set (BSS) in the standard specification. There are two types of BSSs: Independent BSS and Infrastructure BSS. Stations in an independent BSS communicate directly with each other. In contrast, stations in an infrastructure BSS communicate with each other via an Access Point (AP). That is, all traffic to and from a station must flow through the AP, which acts as a base station.

This paper focuses on infrastructure BSSs. We assume that all voice streams are between stations in different BSSs, since users seldom call their neighbors in the same BSS. All voice traffic generated within a BSS is delivered to their called parties located at another BSS.

For illustration, let us consider the network architecture as shown in Fig. 1a. Each AP has two interfaces, an 802.11 interface which is used to communicate with wireless stations, and an Ethernet interface which is connected to the voice gateway. Two gateways for different BSSs are connected through the Internet. The voice gateway is required by the H.323 standard and is used for address translation, call routing for signaling and admission control purposes [1]. All voice packets will go through the gateway before entering the WLAN.

In the subsequent discussion, we will assume that our proposed voice multiplexer resides in the voice gateway. This is purely for the sake of having a concrete reference design for us to expound on the multiplex-multicast concept. In general, the functionality of the voice multiplexer could reside in the voice gateway, a specially-designed AP, or a server between the voice gateway and a general-purpose AP.

Within a BSS, there are two streams for each VoIP session. The uplink stream is for voice originating from the station to the AP. The downlink stream is for voice originating from the other side of the VoIP session to the station, which flows from the remote gateway to the local gateway, and then through the AP to the station.

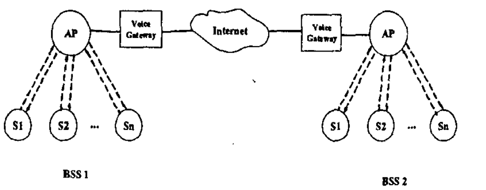

Figure 1a. Traffic Flows in Ordinary VoIP Scheme

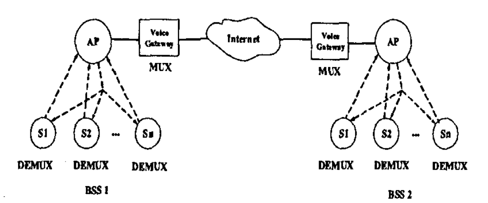

Figure 1b. Traffic Flows in VoIP Multiplex-Multicast Scheme

2.2 Packet Multiplexing and Multicasting

The main idea of our packet multiplex-multicast (M-M) scheme is to combine the data from several downlink streams into a single packet for multicast over the WLAN to their destinations. In this way, the overheads of multiple VoIP packets can be reduced to the overhead of one multicast packet.

The MUX and DEMUX procedures are illustrated in Fig. 2. Specifically, the downlink VoIP traffic first goes through a multiplexer (MUX) in the voice gateway. The MUX replaces the RTP, UDP and IP header of each voice packet with a compressed miniheader, combines multiple packets into a single multiplexed packet, then multicasts the multiplexed packet to the WLAN through the AP using a multicast IP address. All VoIP stations are set to be able to receive the packets on this multicast channel.

The payload of each VoIP packet is preceded by a miniheader in which there is an ID used to identify the session of the VoIP packet. The receiver for which the VoIP packet is targeted makes use of this ID to extract the VoIP packet out of the multiplexed packet. The extraction is performed by a demultiplexer (DEMUX) at the receiver. After retrieving the VoIP payload, the DEMUX then restores the original RTP header and necessary destination information, and assembles the data into its original form before forwarding it to the VoIP application. Other details of context mapping can be found in [2].

All the stations will use the normal unicasting to transmit uplink streams. The AP delivers the upstream packets it receives to the other BSS, whereupon the voice gateway at the other BSS sends the packets to their destinations using the same multiplexing scheme described above. From Fig. 1b, we see that this scheme can reduce the number of VoIP streams in one BSS from $2n$ to $n+1$, where n is the number of VoIP sessions.

The MUX sends out a multiplexed packet every T ms, which is equal to or shorter than the VoIP inter-packet interval. For GSM 6.10, the inter-packet interval is 20 ms. Larger values of T can improve bandwidth efficiency since more packets can be multiplexed, but the delay incurred will also be larger. For example, if T = 10 ms, every two multiplexed packet contains one voice packet from each VoIP stream. The maximum multiplexing time for one voice packet is 10 ms. If T = 20 ms, every multiplexed packet contains one voice packet from each VoIP stream, and the maximum multiplexing time is 20 ms. By adjusting T, one can control the tradeoff between bandwidth efficiency and delay.

Several aspects of VoIP multicasting over WLAN need to be addressed before we conclude this section. The first is the security implication. Since the multicast packets are received by all VoIP stations, a station could then extract VoIP packets not targeted for it and eavesdrop on others' conversations. However, VoIP multicasting over WLAN is no more insecure than regular unicast VoIP over WLAN. One could easily use a sniffer to collect all packets, unicast or multicast, in the WLAN – in fact, there are many free sharewares for doing that. The security problem in both cases should be solved by encrypting the voice packets.

The second aspect is that we have assumed in the above description that there is no additional delay other the MUX delay in the M-M scheme. It should be pointed out that when the power saving mode of 802.11 is turned on at some wireless stations, according to the 802.11 standard, multicast packets for them will be sent out at most only once every beacon period, after DTIM. Waiting for the next beacon will add additional delays to multicast packets. We do not advocate turning on of power saving mode for VoIP stations for this reason. Furthermore, power saving mode is effective only if traffic for the stations arrive at the AP sporadically, which is not the case with VoIP traffic. We have verified through experiments that for commercial products, if the power saving mode is not turned on, multicast packets are sent when they become available, and not after DTIM.

Although the maximum radio rate for 802.11b is 11Mbps, we found that some commercial products (e.g., Lucent Orinoco, Cisco) transmit multicast packet at 2Mbps bit-rate by default. This is due to the nature that in multicasting, the transmitter does not know who the receivers are. For backward compatibility, the sender uses 2 Mbps to transmit multicast packets so that the earlier versions of 802.11 products whose maximum data rate is 2 Mbps can receive them. There is usually a flag in the products to control this backward compatibility. We can simply disable this flag to use 11 Mbps multicast.

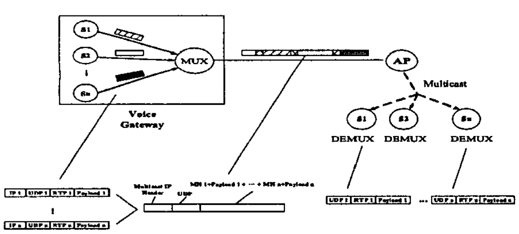

Figure 2. MUX/DEMUX Procedure

2.3 Header Compression

Besides aggregating VoIP streams, we can also increase the bandwidth efficiency by compressing the packet headers during multiplexing. The idea of RTP/UDP/IP header compression comes from two properties in most types of RTP streams. The first is that most of the fields in the IP, UDP and RTP headers do not change over the lifetime of an RTP session. Second, RTP header fields like sequence number and timestamp are increased by a constant amount for successive packets in a stream. So differential coding can be applied to compress these fields into fewer bits.

Our compression is similar to the scheme proposed in [2]. It depends on the use of context-mapping tables in MUX and DEMUX to record necessary information such as RTP header for future reconstruction, source IP address for differentiation between VoIP sessions, synchronization for proper (de)compression and (de)multiplexing. With this scheme, the RTP+UDP+IP header can be replaced with a 2-byte miniheader for most voice packets. We refer the reader to [2] for details. The major reason for the improved efficiency of our system here is the MUX/DEMUX scheme rather than the header compression scheme.

3. Capacity Analysis

In this section, we consider the continuous-bit-rate (CBR) voice sources. For CBR sources, voice packets are generated at the voice codec rate. We focus on the GSM 6.10 codec in this paper, although the general principle we propose is applicable to other codecs as well. For GSM 6.10, the payload is 33 bytes. The time between two adjacent frames is 20 ms, corresponding to a rate of 50 packets per second per VoIP stream. The attributes of other commonly used codecs can be found in [3].

3.1 VoIP Capacity Analysis for 802.11b

Let n be the maximum number of sessions that can be supported. The transmission times for downlink and uplink packets are T_{down} and T_{up}, respectively. Let T_{avg} be the average time between the transmissions of two consecutive packets in a WLAN. That is, in one second, there are totally $1/T_{avg}$ packets transmitted by the AP and all the stations. So,

$$1/T_{avg} = \textit{number of streams} * \textit{number of packets sent by one stream in one second.} \quad (1)$$

Capacity of Ordinary VoIP over WLAN

For a VoIP packet, the header overhead OH_{hdr} consists of the headers of RTP, UDP, IP and 802.11 MAC layer:

$$OH_{hdr} = H_{RTP} + H_{UDP} + H_{IP} + H_{MAC} \quad (2)$$

Besides, at the MAC layer, the overhead incurred at the sender is

$$OH_{sender} = DIFS + averageCW + PHY \quad (3)$$

If it is the unicast packet, the overhead incurred at the receiver is

$$OH_{receiver} = SIFS + ACK \quad (4)$$

where $averageCW = slotTime * (CW_{min} - 1)/2$ is the average backoff time when there are no other contending stations. We ignore the possibility of collisions and the increase of backoff time in subsequent retransmissions after a collision in the analysis here. This means that the VoIP capacity we derive is an upper bound on the actual capacity. However, contention overhead is negligible compared with other overheads, and the analytical upper bound is actually a good approximation of the actual capacity, as will be verified by our simulation results later. So, we have

$$T_{down} = T_{up} = (Payload + OH_{hdr}) * 8 / dataRate + OH_{sender} + OH_{receiver} \quad (5)$$

In the ordinary VoIP case, we have n downlink and n uplink unicast streams. On average, for every downlink packet, there is a corresponding uplink packet. So,

$$T_{avg} = (T_{down} + T_{up}) / 2 \quad (6)$$

From (1), we have

$$1 / T_{avg} = 2n * N_p \quad (7)$$

where N_p is the number of packets sent by one stream per second.

The values of $DIFS, PHY, SIFS, ACK$ are defined by the 802.11b standard [4]. Assuming GSM 6.10 is used, $Payload$ is 33 bytes, N_p is 50. $dataRate$ is 11 Mbps. Solving (7), we get $n = 11.2$. We see that 802.11b WLAN can only support around 11 VoIP sessions from the analysis.

Capacity of Multiplex-Multicast Scheme over WLAN

In this case, the RTP, UDP and IP header of each unmultiplexed packet is compressed to 2 bytes. n packets are aggregated into one packet and they share the same header overhead, which includes UDP, IP and MAC headers of the multiplexed packet. There is no RTP header in the multiplexed packet. In addition, since the multiplexed packet is sent using multicast, it does not have $OH_{receiver}$. So,

$$T_{down} = [(Payload + 2) * n + H_{UDP} + H_{IP} + H_{MAC}] * 8 / dataRate + OH_{sender} \quad (8)$$

Here on average, for one downlink packet, there are totally n corresponding uplink packets. We have

$$T_{avg} = (T_{down} + n * T_{up}) / (n + 1) \quad (9)$$

where T_{up} is the same as (5). Solving (8) and (9) with

$$1 / T_{avg} = (n + 1) * N_p \ , \quad (10)$$

we get $n = 21.2$.

We also derive the capacities when other codecs than GSM 6.10 are used in a similar way, and the results are listed in Table 1. We see that for most of the codecs, the M-M scheme can nearly double the capacity.

Table 1. VoIP Capacities assuming Different Codecs

Codecs	Ordinary VoIP	Multiplex-Multicast Scheme
GSM 6.10	11.2	21.2
G.711	10.2	17.7
G. 723.1	17.2	33.2
G. 726-32	10.8	19.8
G. 729	11.4	21.7

Note that in the above, we assume the average CW wait time to be 15.5 time slots (i.e., $(CW_{min} - 1)/2$). When there is more than one station, the average CW wait time is in fact smaller than this. This accounts for the observation in our simulations (see Table 2) that the maximum session is actually a little bit larger, even though we have ignored the possibility of increase in backoff time due to collisions in our analysis.

3.2 Simulations

We have validated our capacity analysis of 802.11b by simulations. The simulator ns-2 is used. In the simulations, we only consider the local part (BSS1 plus the corresponding voice gateway) of the network shown in Fig. 1a, since our focus is on WLAN, not the Internet. The payload size and frame generation interval are those of the GSM 6.10 codec.

We increase the number of VoIP sessions until the per stream packet loss rate exceeds 1%. We define the system capacity to be the number of VoIP sessions that can be supported while maintaining the packet loss rate of every stream to be below 1%. In our simulations, we assume that the retry limit for each packet is 3. In other words, after a packet is retransmitted three times, it will be discarded regardless of whether the last transmission is successful. Commercial products by Orinoco, for example, adopt a retry limit of 3.

For ordinary VoIP over WLAN, the simulations yield capacities of 12. The result matches the analysis very well. We also tried to increase the number of sessions by one beyond the capacity. We observed that this leads to a large surge in packet losses for the downlink streams. For example, when the 13th session is added, the packet loss rate for downlink streams abruptly jumps to around 6%, while the loss rate for the uplink is still below 1%.

This result is due to the symmetric treatment of all stations in 802.11: the AP is not treated differently from other stations as far as the MAC layer is concerned. For ordinary VoIP over WLAN, the AP needs to transmit *n* times more traffic than each of the other stations. When *n* is smaller than the system capacity, there is sufficient bandwidth to accommodate all transmissions of the AP. However, when *n* exceeds the system capacity, since all stations including the AP are treated the same, the "extra" traffic from the AP will be curtailed, leading to a large packet loss rate for downlink VoIP streams.

This observation provides an alternative explanation as to why the M-M scheme can improve the VoIP capacity. With n VoIP packets multiplexed into one packet, the AP traffic in terms of number of packets per second is reduced to the same as the traffic of each of the other stations.

The results of the M-M scheme are also listed in Table 2. The simulation shows that the capacity can be improved to 22, which matches analysis quite well.

Table 2. Analysis vs. Simulation: Capacity of Ordinary VoIP and Multiplex-Multicast Schemes assuming GSM 6.10 codec

Different Schemes	Analysis	Simulation
Original VoIP	11.2	12
Multiplex-Multicast Scheme	21.2	22

4. Delay Performance

The previous section studied VoIP capacities over WLAN based on a packet-loss rate target of 1%. To provide good voice quality, besides low packet-loss rates, we also need to consider the delay performance. In the following, we present results on the local delays incurred at the voice gateway and the WLAN.

With ordinary VoIP, the access delay within the WLAN is the only local delay. At the AP, the access delay of a VoIP packet is the time between its arrival to the AP until it is either successfully transmitted over the WLAN or dropped at the head of the queue because it has exhausted the retry limit for retransmissions. At the client, the access delay of a VoIP packet is time from when the packet is generated until it leaves the interface card, either due to successful transmission or exhaustion of the retry limit.

With the M-M scheme, in addition to the aforementioned access delay, the local delay for the downlink also includes the MUX delay incurred at the VoIP multiplexer. The MUX delay is the time from the arrival of a VoIP packet to the multiplexer until the time at which the next multiplexed packet is generated. With a multiplexing interval of 20 ms, for example, the MUX delays are distributed between 0 ms and 20 ms.

From an end-to-end viewpoint, it is essential for the local delay to be small so that the overall end-to-end delay of a VoIP stream can be bounded tightly to achieve good quality of service. As a reference benchmark for our delay investigations in this paper, we set a requirement that no more than 1% of the downlink or uplink VoIP packets should suffer a local delay of more than 30 ms. This allows ample delay margin for delay in the backbone network for an end-to-end delay budget of 125 ms [1].

4.1 Access Delay

Figure 3a shows the access delays of successive packets of one randomly chosen CBR VoIP session in the ordinary VoIP scheme when there are 12 simultaneous CBR VoIP sessions (i.e., the system capacity is fully used). The graph on the left is the access delay incurred by the downlink traffic in the AP, while the graph on the right is the access delay incurred by the uplink traffic in its wireless station.

The average delay and delay jitter (defined to be the standard deviation of delay) in the AP are 2.5 ms and 1.4 ms, respectively. The average delay and delay jitter in the wireless station are 1.2 ms and 1.0 ms, respectively. The three-sigma delays (i.e., average delay + 3 * standard deviation) in the AP and wireless stations are therefore 6.7 ms and 4.2 ms, respectively. This means that if the delays were to be normally distributed, less than (1-99.73%) = 0.27% of the packets would suffer local delays larger than 30 ms. Thus, we see that even when the VoIP capacity is fully used, the local delay requirement can be met comfortably.

In addition to delay jitter, we can also look directly at the probability of access delay being smaller than a value. Table 3 tabulates such delay distributions, where A is the random variable representing the access delay. Again, it shows that the requirement of less than 1% of packets having more than 30 ms delay can be met comfortably.

Figure 3b shows the access delay when the M-M scheme is adopted, and the number of VoIP sessions is equal to the previously found capacity of 22. The average delay and delay jitter for the AP (about 0.9 ms and 0.2 ms) and the wireless stations (about 2.0 ms and 1.5 ms) can still comfortably meet the three-sigma metric. From the left side of Fig. 3b, we can see the effect of multicasting downlink packets. Since there are no link layer retransmissions for the packets when collisions occur, the delays at the AP are quite smooth compared with the delays at the client (right side of Fig. 3b), where the uplink VoIP packets are transmitted using unicast. The probability of local delay being less than 30 ms will be presented later in Section 4.2, in which we add the multiplexing delay to the access delay to arrive at the actual local delay in the M-M scheme.

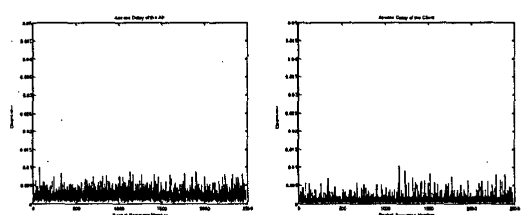

Figure 3a. Access Delays in AP and a Station in Original VoIP over WLAN when there are 12 Sessions

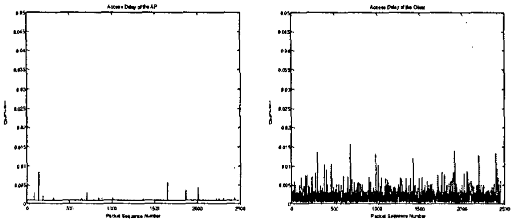

Figure 3b Access Delay in AP and a Station in M-M Scheme when there are 22 Sessions

Figure 3. Delays for CBR VoIP over WLAN

4.2 Extra Delay Incurred by the Multiplex-Multicast Scheme

A VoIP packet will encounter extra delay at the MUX when it waits for the MUX to generate the next multiplexed packet. Recall that the MUX will send off one multiplexed packet to the AP once every T seconds. Since we set the multiplexing period to be at most one audio-frame period in our study, our scheme ensures that the extra delay incurred at the MUX is bounded by one frame period (20 ms if GSM 6.10 codec is used). Note that only downlink packets go through the MUX.

To account for the extra delay, we define M to be the random variable representing the extra multiplexing delay. We assume M to be uniformly distributed between 0 and 20 ms. Table 4 tabulates the distribution of multiplexing plus access delays incurred at the AP and the distribution of access delay incurred at the wireless stations. As shown, the local delay budget of 30 ms can be met comfortably.

Table 3. Access Delay Distribution for Ordinary VoIP when System Capacity of 12 is Fully Used

	Access delay for the AP (Local delay for downlink VoIP packets)	Access delay for the station (Local delay for uplink VoIP packets)
$\Pr[A \le 0.01s]$	1	0.999
$\Pr[A \le 0.03s]$	1	1
$\Pr[A \le 0.05s]$	1	1

Table 4. Delay Distributions for Multiplex-Multicast Scheme when System Capacity of 22 is Fully Used

Access delay for the AP plus MUX delay in the MUX (Local delay for the downlink VoIP packets)		Access delay for the station (Local delay for the uplink VoIP packets)	
$\Pr[M + A \le 0.01s]$	0.455	$\Pr[A \le 0.01s]$	0.996
$\Pr[M + A \le 0.02s]$	0.955	$\Pr[A \le 0.02s]$	1
$\Pr[M + A \le 0.03s]$	1	$\Pr[A \le 0.03s]$	1

The delay results in this section show that the VoIP capacity we defined in the previous section using the loss metric can also meet the delay metric defined in this section. The Quality of Servive (QoS) of VoIP in terms of loss rate and delay is good enough for both ordinary VoIP and M-M VoIP.

5. Conclusions

This paper has proposed a Multiplex-Multicast (M-M) scheme for VoIP over WLAN. Our scheme aggregates downlink voice packets with header compression in the voice gateway, then multicasts the multiplexed packet to all the stations. The scheme can reduce the large overhead when VoIP traffic is delivered over WLAN. Unlike other VoIP capacity improvement schemes reported in the literature, the M-M scheme requires no changes to the MAC protocol at the wireless end stations. This feature makes our scheme more readily deployable over the existing network infrastructure.

To test our proposed scheme, we set a performance target of i) no more than 1% VoIP packets can be lost; ii) no more 1% of the VoIP packets can experience more than 30 ms overall delay within the WLAN equipment and components introduced by our solutions. The results show that our proposed scheme can achieve a voice capacity nearly 100% higher than ordinary VoIP, while meeting our performance target.

What we can conclude from the results of this paper is this. In an enterprise environment, the number of simultaneous VoIP sessions with the regular scheme will probably have to be reduced to around 5 for 802.11b to make room for traffic of other applications. This may result in an unacceptably high blocking probability for VoIP in many situations. Our Multiplex-Multicast scheme is one way to increase the VoIP capacity and decrease the blocking probability.

References

[1] Bill Douskalis, *IP Telephony, the integration of robust VoIP services*, Prentice Hall PTR, 2000.

[2] Sze, H.P., Liew, S.C., Lee, J.Y.B. and Yip, D.C.S, "A multiplexing scheme for H.323 voice-over-IP applications", *IEEE J. Select. Areas Commun*, Vol. 20, pp. 1360-1368, Sept. 2002.

[3] T. Hoshi, K. Tanigawa, and K. Tsukada, "Proposal of a method of voice stream multiplexing for IP telephony systems," in *Proc. IWS'99*, Feb. 1999, pp. 182-188.

[4] Matthew S. Gast, *802.11 Wireless Networks, the definitive guide*, O'REILLY, 2002.

[5] Philip K. McKinley and Suraj Gaurav, "Experimental evaluation of forward error correction on multicast audio streams in wireless LANs", *Proceedings of the eighth ACM international conference on Multimedia*, pp. 416-418, 2000.