Solutions to Performance Problems in VoIP over 802.11 Wireless LAN [1]

Wei Wang, Soung C. Liew
Department of Information Engineering
The Chinese University of Hong Kong

Victor O. K. Li
Department of Electrical and Electronic Engineering
The University of Hong Kong

Abstract

VoIP over WLAN is poised to become an important Internet application. However, two major technical problems that stand in the way are 1) low VoIP capacity in WLAN; 2) unacceptable VoIP performance in the presence of coexisting traffic from other applications. With each VoIP stream typically requiring less than 10 Kbps, an 802.11b WLAN operated at 11 Mbps could in principle support more than 500 VoIP sessions. In actuality, no more than a few sessions can be supported due to various protocol overheads (For GSM 6.10, it is about 12). This paper proposes and investigates a scheme that can improve the VoIP capacity by close to 100% without changing the standard 802.11 CSMA/CA protocol. In addition, we show that VoIP delay and loss performance in WLAN can be compromised severely in the presence of coexisting TCP traffic, even when the number of VoIP sessions is limited to half its potential capacity. A touted advantage of VoIP over traditional telephony is that it enables the creation of novel applications that integrate voice with data. The inability of VoIP and TCP traffic to coexist harmoniously over the WLAN poses a severe challenge to this vision. Fortunately, the problem can be largely solved by simple solutions that require only changes to the MAC protocol at the Access Point. Specifically, in our proposed solutions, the MAC protocol at the wireless end stations needs not be modified, making the solutions more readily deployable over the existing network infrastructure.

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,

[1] 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).

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.

Thanks to the convergence of these two trends, we believe VoIP over WLAN is poised to become an important Internet application. Before that can happen, however, two technical problems need to be solved. The first is that the system capacity for voice can be quite low in WLAN. The second is that VoIP traffic and data traffic from traditional applications such as Web, e-mail, etc., can interfere with each other and bring down VoIP performance.

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 μsec. The transmission time for the 40-byte IP/UDP/RTP header is 40 * 8 / 11 = 29 μsec. However, the 802.11 MAC/PHY layers have additional overhead of more than 800 μsec, 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%.

In an enterprise WLAN or public WLAN hotspot, supporting VoIP becomes even more complicated, since the WLAN needs to simultaneously support other applications besides VoIP. Providing room for these applications may further limit the number of VoIP sessions. As will be shown later in this paper, even when the number of VoIP sessions is limited to just half of the capacity in an 802.11b WLAN, interference from just one TCP connection will cause unacceptably large increases in the delay and packet-loss rate of VoIP traffic.

The investigations of this paper revolve around finding solutions for the two fundamental problems above. We focus our attention on solutions that do not require modifications on the 802.11 hardware and firmware at the client stations so that they can be more readily deployed. Specifically, the contributions of this paper are as follows:

1) We propose a voice multiplex-multicast (M-M) scheme for overcoming the large overhead effect of VoIP over WLAN. The M-M scheme eliminates inefficiency in downlink VoIP traffic by multiplexing packets from several VoIP streams into one multicast packet for transmission over the WLAN. The net result is that the overhead of the multicast packet is shared by many constituent VoIP packets.
2) We have conducted comprehensive performance studies on the M-M scheme. Our studies include several popular voice codecs, CBR and VBR voice encoding, and 802.11b, 802.11a, and 802.11g MAC protocols. The results show that the M-M scheme can achieve a voice capacity 80% to 90% higher than ordinary VoIP over WLAN. In addition, the delay incurred by the M-M scheme is well below 125 $m\sec$, leaving ample delay margin for the backbone network as VoIP packets travel from one WLAN to another WLAN.
3) We demonstrate the inherent interference problem between VoIP and TCP traffic at the buffer of the AP (Access Point) in a WLAN, and use a simple priority queuing solution that effectively eliminates the problem. This solution is effective in both the M-M and ordinary VoIP setups.
4) Last by not least, in our investigation of the M-M scheme under the interference of unicast traffic, we found the loss rate of multicast packets to be excessive when there are upstream TCP packets due to packet collisions. The reason is that unlike for unicasting, there is no ARQ for multicasting at the MAC layer of 802.11, and collided multicast packets are not retransmitted. Excessive multicast packet loss due to collisions is a fundamental problem in WLAN that has no parallel in the Ethernet [2]. We provide and demonstrate the effectiveness of a simple solution that solves this unreliability problem of WLAN multicasting *in general*.

In the following discussions, we assume a perfect channel condition, which means that there are no transmission error and link adaptations. Based on our own real experiments, within a reasonable range, the actual packet loss rate is negligible. The detailed experimental results and explanations are given in Section 7.

[2] Note that this problem does not occur in the regular Ethernet, in which collisions of multicast packets can be detected by the sender itself and the packets can be retransmitted. Collision detection by the sender while it is transmitting is technically difficult in radio networks. In 802.11, the sender relies on the receiver to return an ACK after it has received a packet. If an ACK is not returned immediately, the sender deduces that the packet has been lost. While this is a good *indirect* way to detect collisions for unicasting, it is not viable for multicasting in which receivers are free to join or leave a multicast group without informing the sender. It is for this reason that there is no ACK mechanism for multicasting in 802.11. Unfortunately, this makes multicasting much more unreliable in the WLAN than in the Ethernet.

2. Background

2.1 VoIP Attributes

For VoIP, the analog or PCM voice signals are encoded and compressed into a low-rate packet stream by codecs. Table 1 lists the attributes of several commonly used codecs. Generally, the codecs generate constant bit-rate audio frames consisting of 40-byte IP/UDP/RTP headers followed by a relatively small payload. 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.

Table 1. Attributes of Commonly Used Codecs

Codec	GSM 6.10	G.711	G.723.1	G.726-32	G.729
Bit rate (Kbps)	13.2	64	5.3/6.3	32	8
Framing interval (ms)	20	20	30	20	10
Payload (Bytes)	33	160	20/24	80	10
Packets /sec	50	50	33	50	50*

* For all codecs except G.729, Packets/sec = 1 / (Framing interval). For G.729, two frames are combined into one packet so that Packets/sec = 1/(2* Framing interval)

2.2 IEEE 802.11

There are two access mechanisms specified in the IEEE 802.11 standard: Distributed Coordination Function (DCF) and Point Coordination Function (PCF). PCF is a centralized mechanism, where one central coordinator polls other stations and allows them contention free access to the channel. However, PCF is an option not supported in most commercial products.

DCF is based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol. The basic operation of 802.11 DCF is described in Fig. 1. Before transmission, a station will randomly choose a backoff time with number of time slots ranging from 0 to Contention Window (CW) -1. The station will decrease the backoff-timer counter progressively while the channel is idle after a DCF Inter Frame Space (DIFS) and pause the timer if it senses the channel to be busy. When the backoff value reaches zero, the station will transmit its packet.

If this is a unicast packet, the station will wait for the receiver to send back an ACK frame after a Short Inter Frame Space (SIFS) interval. If it does not receive the ACK, the station assumes the packet has been lost due to transmission errors or a collision. Thereafter, it doubles the CW value, generates a backoff time chosen randomly from the interval [0, CW-1], and retransmits this packet following the same procedure as above.

For a multicast or broadcast packet, the transmitting station will not wait for the ACK, as multicast receivers do not send back ACKs in general. There are no retransmissions

for multicast and broadcast packets in 802.11 DCF. The station will proceed to send the next packet regardless of whether the earlier packet has been received successfully.

The values of the parameters of 802.11b DCF are listed in Table 2.

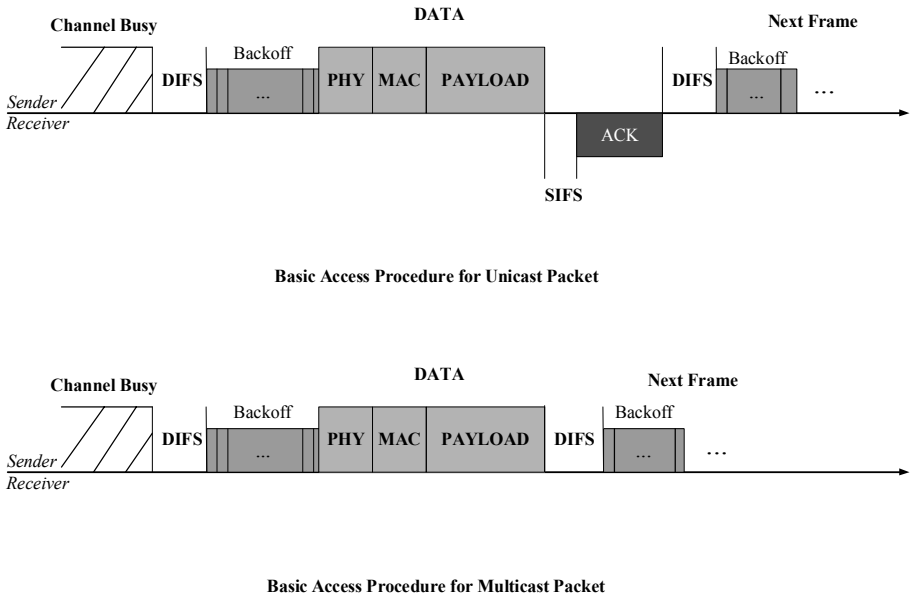

Figure 1. Basic Operation of 802.11 DCF

Table 2. Parameter Values of 802.11b DCF

DIFS	50 μsec
SIFS	10 μsec
Slot Time	20 μsec
CWmin	32
CWmax	1023
Data Rate	1, 2, 5.5, 11 Mbps
Basic Rate	2 Mbps
PHY header*	192 μsec
MAC header	34 bytes
ACK*	248 μsec

* PHY header is transmitted at 1 Mbps, ACK shown above is actually ACK frame + PHY header. The ACK frame is 14 bytes and is transmitted at basic rate, 2 Mbps, regardless of the data rate.

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.

2.3 Related Work

Previous work on VoIP over WLAN can be classified according to which access mechanism, DCF or PCF, is used. References [3] and [4] assumed the use of PCF. However, as mentioned above, PCF is not supported in most 802.11 products, and its popularity pales in comparison to DCF. A reason could be that the market does not see a compelling need for PCF. In addition, DCF is a technology that has been well tested and proven to be robust in the field. For example, when there are two overlapping WLANs using the same frequency channel, DCF will continue to work while PCF will not, since collisions between stations of the two WLANs may occur during their supposedly contention-free periods.

References [5 – 9] studied the use of DCF to support VoIP. Specifically, results in [6] and [7] confirm the existence of similar capacity limits as identified in this paper. However, no solutions are provided to improve the VoIP capacity over WLAN. References [5], [8] and [9] investigated various schemes for improving the VoIP capacity, but all the proposed schemes require modifications of the MAC protocol used by the VoIP stations. Reference [9] has an even more stringent requirement that the MAC protocol of the non-VoIP data stations must also be modified. In contrast, our M-M scheme requires no changes to 802.11 MAC layers of the VoIP and non-VoIP stations. In addition, our solutions that allow harmonious co-existence of VoIP and TCP require only minor modifications of the AP MAC layer.

There have been many schemes proposed for reliable multicast in general [10 – 12]. Most of them attempt to achieve 100% reliability by using some sort of retransmission strategies, at the expense of delay. Such approaches are not scalable and may cause VoIP to have unacceptable delay. Zero packet loss rate is too stringent a requirement for VoIP and is not necessary. Our paper demonstrates a simple scheme that solves the main cause for multicast packets losses in WLAN, namely, packet collisions. Specifically, in scheduling the transmission of multicast packets, our scheme 1) replaces DIFS with a Multicast Inter Frame Space (MIFS), with SIFS < MIFS < DIFS; 2) set the contention window, CW, to 1. This solution can in principle be incorporated into mechanisms provided by the newly proposed 802.11e standard.

3. VoIP Multiplex-Multicast Scheme

3.1 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. 2a. 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.

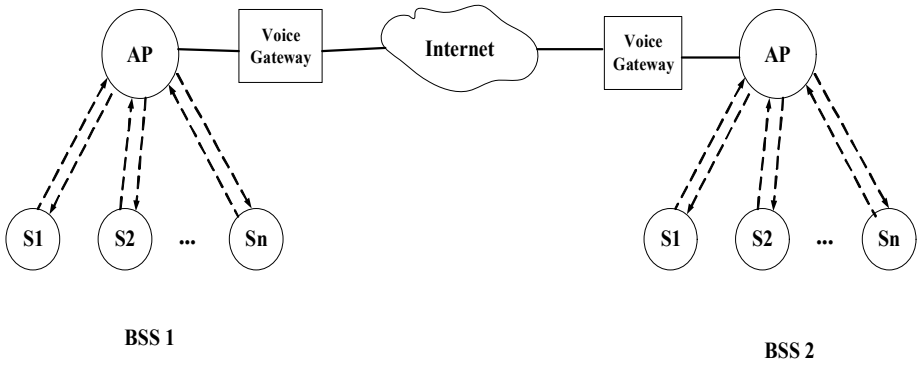

Figure 2a. Traffic Flows in Ordinary VoIP Scheme

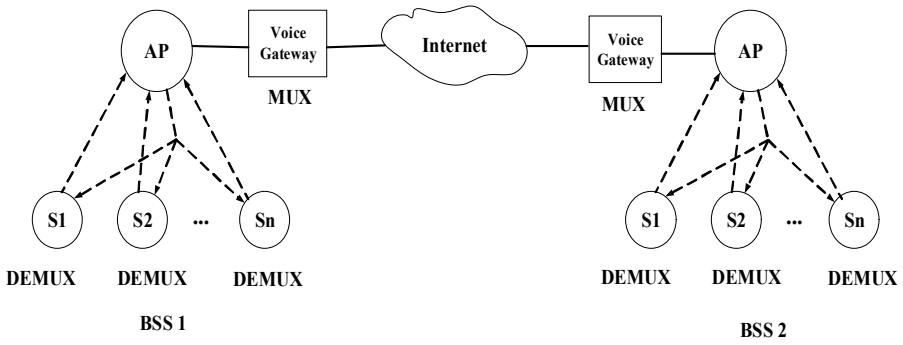

Figure 2b. Traffic Flows in VoIP Multiplex-Multicast Scheme

3.2 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. 3. 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 [13].

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. 2b, 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.

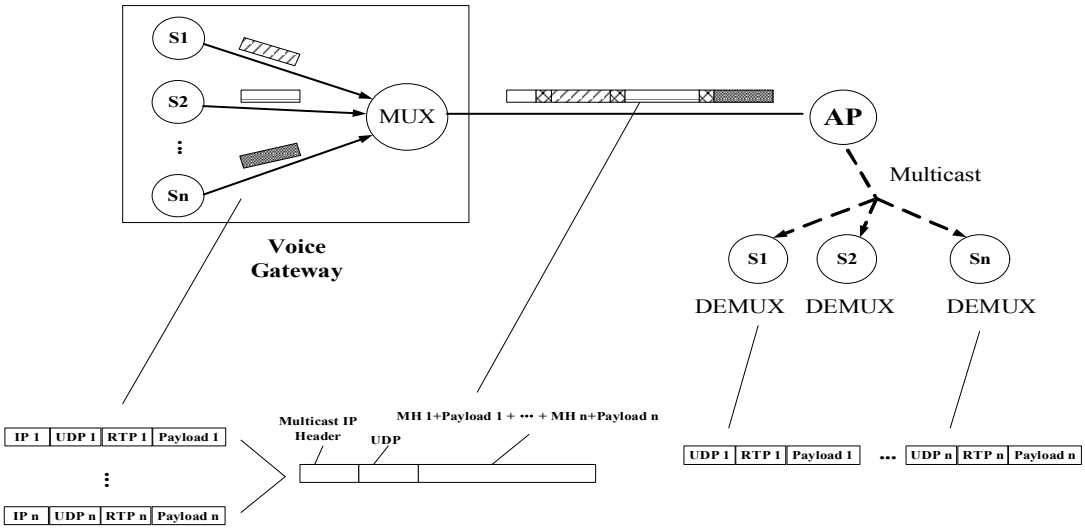

Figure 3. MUX/DEMUX Procedure

Two 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.

3.3 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 [13]. 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 [13] for details. The major reason for the improved efficiency of our system here is the MUX/DEMUX scheme rather than the header compression scheme.

4. Capacity Analysis

In this section, we consider both continuous-bit-rate (CBR) and variable-bit-rate (VBR) voice sources. For CBR sources, voice packets are generated at the voice codec rate (e.g., 50 packets per second when GSM 6.10 is used). We model VBR sources using the Brady's ON-OFF model [19], in which data is generated at the voice codec rate during the ON state, and no data is generated during the OFF state. As in [19], we assume the ON and OFF times to be exponentially distributed with means of 1 sec and 1.35 sec, respectively. We first consider the CBR case in the following capacity analysis.

4.1 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,

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

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 $DIFS, PHY, SIFS, ACK$ for 802.11b are listed in Table 2. 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 3. We see that for most of the codecs, the M-M scheme can nearly double the capacity.

Table 3. 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 6) 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.

4.2 VoIP Capacity Analysis for 802.11a and 802.11g

802.11a uses the same MAC protocol as 802.11b but with a different set of parameters. In 802.11a, the PHY preamble and the contention time slot are shorter, and the maximum data rate is much larger (see Table 4). Therefore, 802.11a may have a higher system capacity for VoIP. 802.11a, however, is not compatible with 802.11b.

802.11g also has the same maximum data rate as 802.11a. However, it has two different operation modes. In the 802.11g-only mode, all stations in the WLAN are 802.11g stations, so that they can operate in a way that is more efficient but not compatible with 802.11b. In the 802.11b-compatible mode, some stations in the WLAN are 802.11b stations, and 802.11g stations must operate in a way that is compatible with 802.11b.

In the 802.11g-only mode, timing spaces even smaller than those in 802.11a are used (Table 4), leading to a slightly higher capacity than 802.11a. However, the use of 802.11g-only mode in practice is unlikely given the large installed base of 802.11b equipment already in use. After all, the main motivation for the use of 802.11g over 802.11a is that 802.11g is compatible with 802.11b while 802.11a is not. One would expect 802.11g stations to mostly operate in the 802.11b-compatible mode in the field.

Although in the 802.11b-compatible mode of 802.11g, the maximum data rate of 54 Mbps is much larger than the 11 Mbps of 802.11b, the other overheads are comparable. For packets with large payload, higher throughput than that in 802.11b can be achieved. Unfortunately, VoIP packets have very small payload. As a result, the higher data rate of 54 Mbps does not yield much improvement as far as VoIP capacity is concerned, since the dominant overheads are not reduced. The following paragraph elaborates the operation of the 802.11b-compatible mode.

In the 802.11b-compatible mode, the DIFS, SIFS and contention slot time are the same as those in 802.11b, so that 802.11g and 802.11b stations can contend for the access of the channel in a fair manner. Furthermore, 802.11g has to enable “protection”, wherein the 802.11g stations operating at the higher data rate must reserve the channel before accessing it at the higher speed using a slower reservation mechanism understandable by the 802.11b stations.

There are two kinds of protections. The first is CTS-to-self, in which an 802.11g station needs to send a Clear-To-Send (CTS) frame to clear the channel before transmitting a data frame. This CTS frame is sent at the 802.11b basic rate using the 802.11b PHY preamble so that 802.11b stations as well as other 802.11g stations can hear it. The NAV value in the CTS frame specifies how long the channel will be reserved. The CTS-to-self mode is not targeted for solving the hidden node problem. For that, the

RTS-CTS protection mode is used, in which the receiving station must return an RTS frame after the CTS frame before the transmitting station begins transmission.

Table 4. Parameter Values of 802.11a and 802.11g

	802.11a	802.11g	
		802.11g-only	802.11b-compatible
DIFS	34 us	28 us	50 us
SIFS	16 us	10 us	10 us
Slot Time	9 us	9 us	20 us
CWmin	16	16	16
RTS	14 bytes	14 bytes	14 bytes
CTS	14 bytes	14 bytes	14 bytes
Supported Data Rates	6, 9, 12, 18, 24, 36, 48, 54 Mbps	6, 9, 12, 18, 24, 36, 48, 54 Mbps	1, 2, 5.5, 11, 6, 9, 12, 18, 24, 36, 48, 54 Mbps
Basic Rate	N/A	N/A	2 Mbps
PHY for protection frames *	N/A	N/A	192 us
PHY for other frames	20 us	20 us	20 us
ACK frame	24 us	24 us	24 us

* Protection frames are RTS, CTS frames used in 802.11b-compatible mode of 802.11g

Using the parameters listed in Table 4, we have performed the capacity analysis for 802.11a and 802.11g based on essentially the same set of equations as in the previous section. The results for GSM 6.10 codec with CBR voice source are listed in Table 5.

The analysis for 802.11a and 802.11g is based on several supported data rates. Note that, in practice, different data rates are based on different modulation schemes in the standards (i.e., QAM 64 for 54 and 48 Mbps, QAM 16 for 36 and 24 Mbps, QPSK for 18 and 12 Mbps, BPSK for 9 and 6 Mbps). For the same SNR, different modulation schemes may have different bit error rates (BER). In other words, different data rates may have different coverage areas. Normally, the higher the data rate, the smaller the coverage area. So in the real scenario, 54 Mbps data rate for 802.11a and 11g may not be very reasonable because the coverage area is very small. When the client and the AP are not close enough, the auto rate fallback (ARF) function in the commercial products will tune the data rate to a lower level so as to increase the coverage area.

As expected, 802.11g-only mode can achieve even higher capacities than 802.11a, thanks to its smaller DIFS and SIFS. However, when 802.11g needs to be compatible with 802.11b, the capacity decreases drastically. In particular, when 802.11g adopts RTS-CTS protection, the capacity is not much higher than that in 802.11b. This shows that the higher data rate of 802.11g fails to bring about a corresponding higher VoIP capacity if compatibility with 802.11b is to be maintained.

Two observations need to be pointed out: 1) Given a transmission mode, the capacity does not decrease much as the data rate decreases. For example, for the 802.11g with CTS-to-self protection mode, even the data rate decreases from 54 Mbps to 18 Mbps, the capacity for ordinary VoIP only decreases one and the capacity for M-M scheme only decreases around three. This is because the change of data rate only affects the transmission time of the payload, which only corresponds to a small proportion of the total transmission time of a frame. The major part, such as PHY, Backoff, IFS and ACK do not change with the data rate. 2) For the various data rates of 802.11, the M-M scheme can achieve roughly the same percentage of improvement in VoIP capacity. That is, an improvement of slightly less than 100% for all cases.

Table 5. VoIP Capacities for 802.11b, 802.11a and 802.11g Derived from Analysis

MAC	Ordinary VoIP	Multiplex-Multicast Scheme	Percentage Improved
802.11b (11 Mbps)	11.2	21.2	89.3%
802.11a (54 Mbps)	56.4	108.8	92.9%
802.11a (36 Mbps)	53.9	102.9	90.9%
802.11a (18 Mbps)	47.8	88.4	84.9%
802.11g-only (54 Mbps)	60.5	116.5	92.6%
802.11g-only (36 Mbps)	57.7	109.7	90.1%
802.11g-only (18 Mbps)	50.7	93.4	84.2%
802.11g with CTS-to-self protection (54 Mbps)	18.9	36.6	93.7%
802.11g with CTS-to-self protection (36 Mbps)	18.6	35.9	93.0%
802.11g with CTS-to-self protection (18 Mbps)	17.9	33.9	89.4%
802.11g with RTS-CTS protection (54 Mbps)	12.7	24.3	91.3%
802.11g with RTS-CTS protection (36 Mbps)	12.5	24.0	92.0%
802.11g with RTS-CTS protection (18 Mbps)	12.2	23.1	89.3%

4.3 VoIP Capacity with VBR Sources

VBR encoding can reduce the traffic of VoIP streams so that the capacity for VBR VoIP will be larger in WLAN. For Brady's VBR model, the assumed mean ON time is 1 second, and the mean OFF time is 1.35 second. On average, the traffic load of VBR is $ON/(ON+OFF) = 42.5\%$ of the traffic load of CBR. The VBR VoIP capacity is simply

$$C_{VBR} = C_{CBR} / \rho \tag{11}$$

where C_{CBR} is the capacity for CBR source, $\rho = ON/(ON + OFF) = 42.5\%$. The ordinary VBR VoIP capacity is $11.2/42.5\% = 26.3$, and the Multiplex-Multicast VBR VoIP capacity is $21.2/42.5\% = 49.8$.

4.4 Simulations

We have validated our capacity analysis of 802.11b by simulations. The simulator ns-2 [20] is used. In the simulations, we only consider the local part (BSS1 plus the corresponding voice gateway) of the network shown in Fig. 2a, 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 and 25 for CBR and VBR, respectively. These results match 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, for CBR, when the 13^{th} 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 6. The simulation shows that the CBR capacity can be improved to 22, which matches analysis quite well. However, the VBR capacity can only be improved to 36, which is far below the result of analysis. This can be explained as follows.

Recall that in the analysis we have ignored collisions. For CBR sources, the generated traffic is smooth and collision probability does not go up drastically as the number of

VoIP sessions increase. In fact, the collision probability remains negligible right up to the capacity limit. However, for VBR sources, the traffic is bursty. Our analysis for VBR was based on the average traffic load. But the actual "instantaneous" traffic load fluctuates over time, depending on the number of ON sources. Even when the average traffic load is well below capacity, the instantaneous traffic load could reach a level beyond the throughput limit of WLAN to cause high collision probability.

Thanks to link-layer ARQ, unicast frame can tolerate several collisions before being discarded. The lack of ARQ in WLAN multicast, however, means that multicast frames will be dropped after the first collision. So when our M-M scheme is applied on VBR sources, the capacity is actually limited by the higher propensity for collision loss of downlink multicast frames. Fortunately, we can solve it by applying a minor modification on the AP MAC layer to reduce the collision probability of multicast frames. The details of the modification will be presented in Section 6, in which the same method is used to reduce of collisions of downstream multicast packets with upstream TCP packets. This modification allows the M-M VBR VoIP scheme to have capacity of 46, which is closer to the analytical result in Table 6.

Table 6. Analysis vs. Simulation: Capacity of Ordinary VoIP and Multiplex-Multicast Schemes assuming GSM 6.10 codec

Different Schemes	CBR		VBR	
	Analysis	Simulation	Analysis	Simulation
Original VoIP	11.2	12	26.3	25
Multiplex-Multicast Scheme	21.2	22	49.8	36*

* After applying the method proposed in Section 6, the capacity is actually 46 with loss and delay metric

5. 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 [2].

5.1 Access Delay

Figure 4a 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 cumulative access delay distribution. Figure 5 plots the delay distributions. In addition, Table 7 tabulates the delay distribution in another way to make things clearer, where A is the random variable representing the access delay. Again, they show that the requirement of less than 1% of packets having more than 30 ms delay can be met comfortably.

Figure 4b 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. 4b, 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. 4b), 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 5.2, in which we add the multiplexing delay to the access delay to arrive at the actual local delay in the M-M scheme.

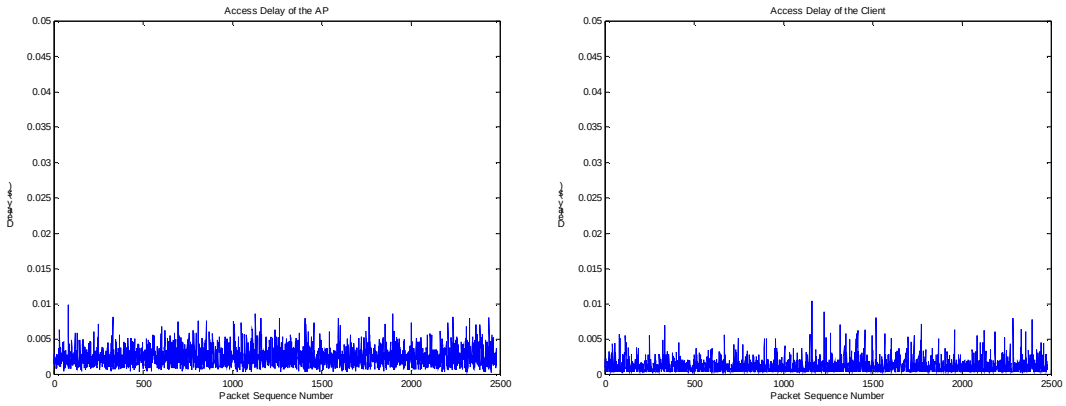

Figure 4a. Access Delays in AP and a Station in Original VoIP over WLAN when there are 12 Sessions

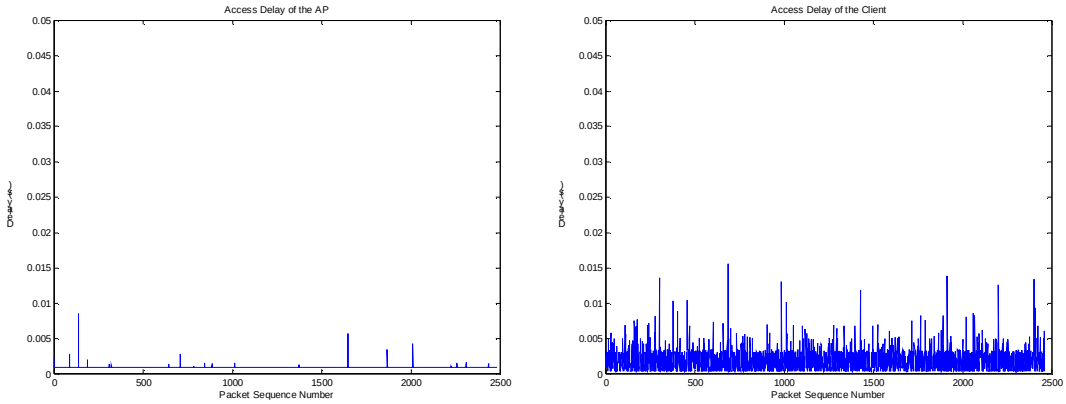

Figure 4b Access Delay in AP and a Station in M-M Scheme when there are 22 Sessions

Figure 4. Delays for CBR VoIP over WLAN

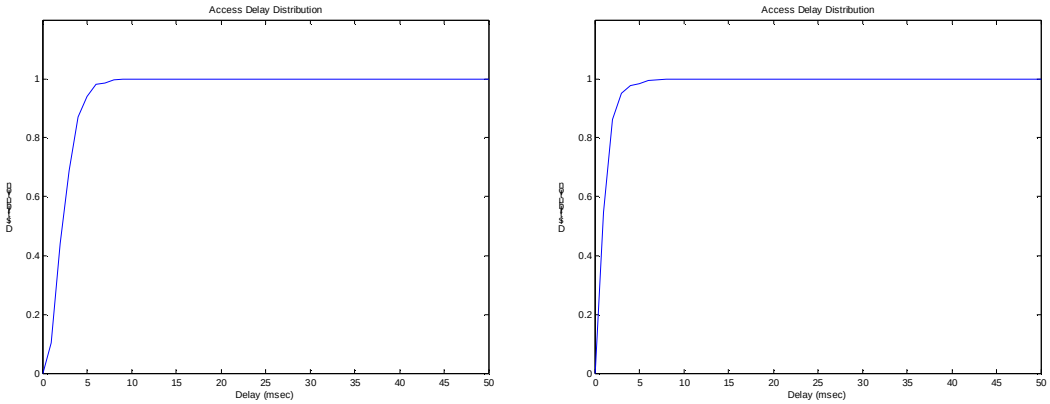

Figure 5a. Cumulative Access Delays Distributions in AP and a Station in Original VoIP over WLAN when there are 12 Sessions

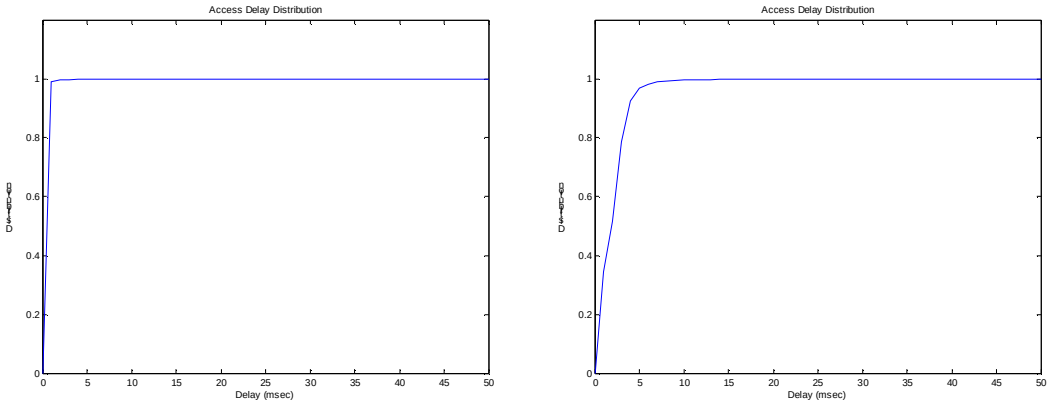

Figure 5b Cumulative Access Delay Distributions in AP and a Station in M-M Scheme when there are 22 Sessions

Figure 5. Cumulative Delay Distributions for CBR VoIP over WLAN

We now look at the performance when VBR encoding is used. Figure 6 plots the delays for successive packets. Figure 7 is the cumulative delay distributions for the same set of data. Figure 6a shows the delay of ordinary VBR VoIP over WLAN. The average delay and jitter for AP (about 3.6 ms and 5.9 ms) and those of the wireless station (about 1.4 ms and 1.3 ms) are still acceptable. However, even though the AP delay meets the three-sigma metric, we find that 1% of the downlink packets have delays larger than 30 ms (see Table 7). This is because the delay is not normally distributed due to the burstiness of the traffic.

Figure 6b shows the delay of the M-M scheme for VBR VoIP when there are 36 sessions. The average delay and delay jitter for AP are 1.1 and 0.7 ms, respectively, and those for the station are 0.9 and 0.7 ms, respectively. The low values of the delay figures suggest that the channel is not fully utilized. Recall that the system capacity of 36 sessions was derived from our simulation results in which we required the packet loss rate to be less than 1%. The results from Fig. 6b show that the capacity is limited by that

loss-rate requirement rather than the delay requirement, and in principle the capacity can be increased if a way can be found to lower the loss rate. Section 6 will consider one such solution.

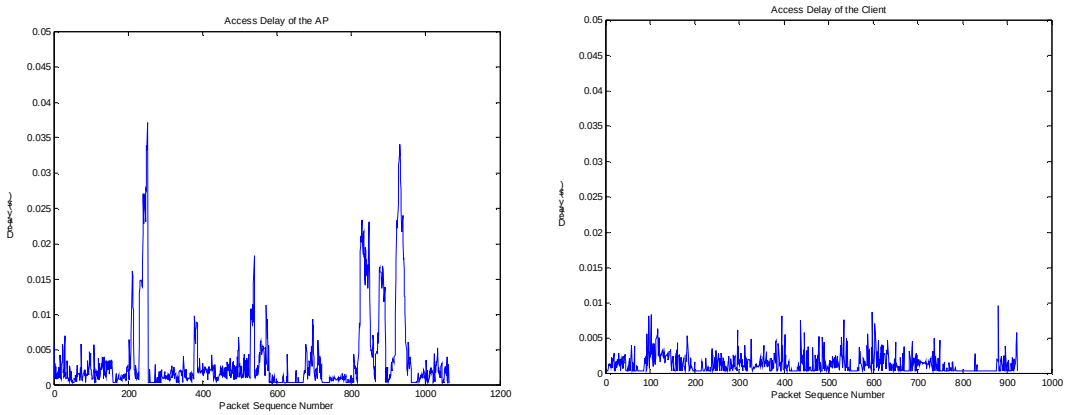

Figure 6a Access Delay in AP and a Station in Original VoIP over WLAN when there are 25 Sessions

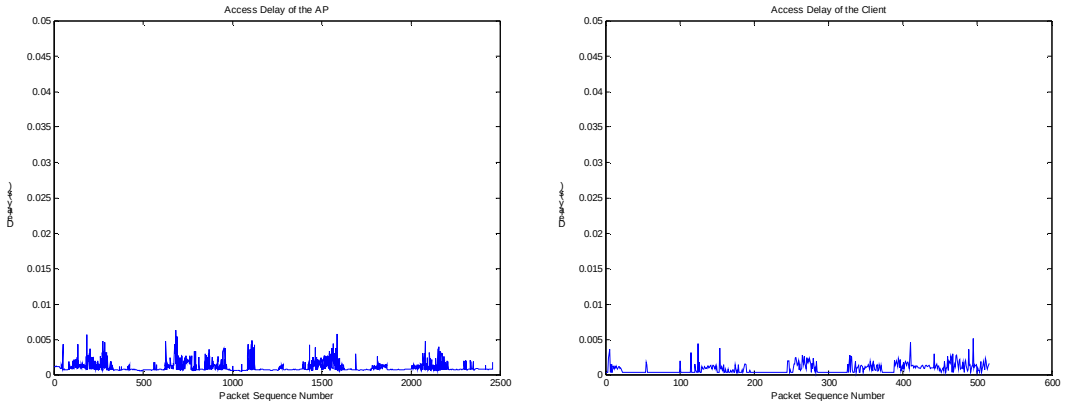

Figure 6b Access Delay in AP and a Station in M-M Scheme when there are 36 Sessions

Figure 6. Delays for VBR VoIP over WLAN

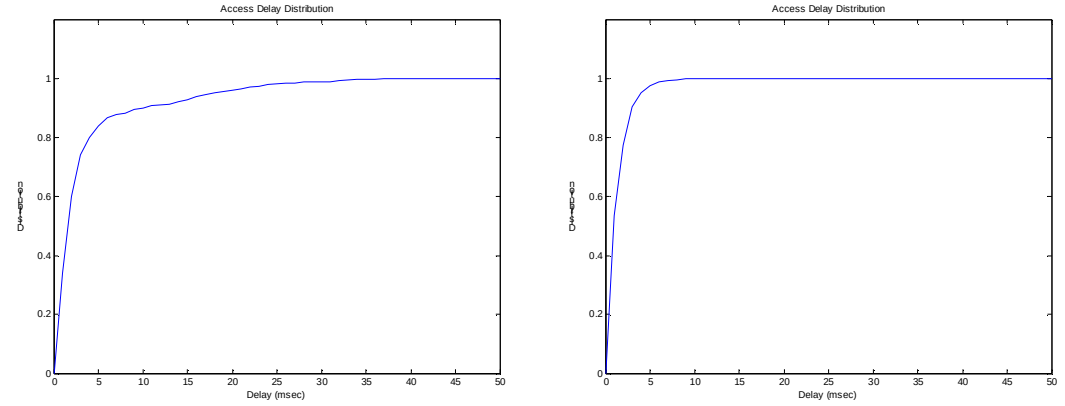

Figure 7a Cumulative Access Delay Distributions in AP and a Station in Original VoIP over WLAN when there are 25 Sessions

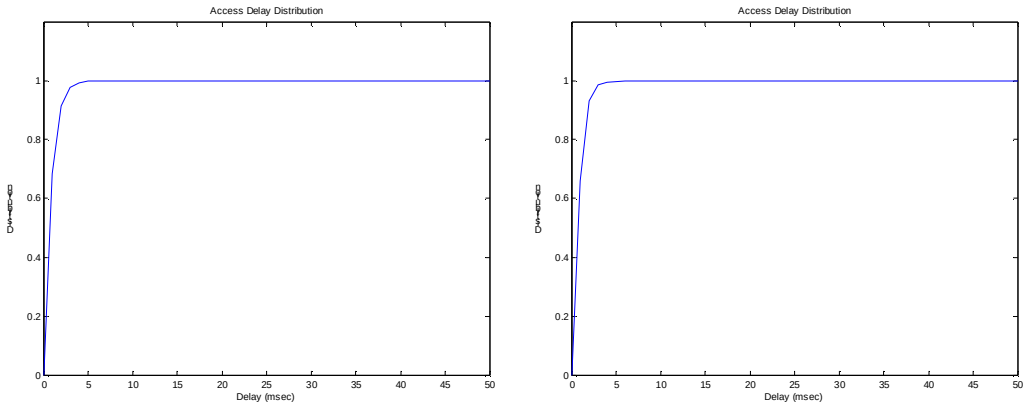

Figure 7b Cumulative Access Delay Distributions in AP and a Station in M-M Scheme when there are 36 Sessions

Figure 7. Cumulative Delay Distributions for VBR VoIP over WLAN

Table 7. Access Delay Distribution for Ordinary VBR VoIP when System Capacity 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)	
	CBR(12)	VBR(25)	CBR(12)	VBR(25)
$\Pr[A \le 10ms]$	1	0.900	0.999	1
$\Pr[A \le 30ms]$	1	0.990	1	1
$\Pr[A \le 50ms]$	1	1	1	1

5.2 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 8 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 for both CBR and VBR VoIP.

Table 8. Delay Distributions for Multiplex-Multicast Scheme when System Capacity is Fully Used

Access delay for the AP plus MUX delay in the MUX (Local delay for the downlink VoIP packet)			Access delay for the station (Local delay for the uplink VoIP packet)		
	CBR(22)	VBR(36)		CBR(22)	VBR(36)
$\Pr[M + A \le 0.01s]$	0.455	0.447	$\Pr[A \le 0.01s]$	0.996	1
$\Pr[M + A \le 0.02s]$	0.955	0.947	$\Pr[A \le 0.02s]$	1	1
$\Pr[M + A \le 0.03s]$	1	1	$\Pr[A \le 0.03s]$	1	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. When there is no other non-VoIP traffic, 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.

6. VoIP Co-existing with TCP Interference Traffic

We have so far considered VoIP without other co-existing traffic in the WLAN. In an enterprise WLAN or public WLAN hotspot, VoIP will likely coexist with traffic from other applications. This traffic is mostly transported using TCP. To make room for the TCP traffic, the number of VoIP sessions should be limited to below the VoIP capacity derived in the previous sections. In addition, the fluctuations of the TCP traffic will also affect the QoS of VoIP. We will only present the results of CBR voice sources in this section. The experimental results for VBR voice sources are similar qualitatively.

6.1 Ordinary VoIP co-existing with TCP over WLAN

Problems Caused by TCP Interference

TCP can interfere with VoIP in two ways. The first occurs at the AP for TCP and VoIP downlink traffic, and the second occurs when traffic at different nodes contend to access the WLAN.

In most commercial APs, all downlink traffic shares a common FIFO queue. In this case, VoIP packets intermix with TCP packets in the AP buffer, leading to the typical UDP/TCP competition problem as pointed out by Floyd [21]. Specifically, delay-insensitive TCP traffic may prevent timely transmission of VoIP data.

TCP generates two-way traffic in the WLAN. After the sender's TCP_DATA packets must be acknowledged by receiver's TCP_ACK packets. In the WLAN, both TCP_DATA and TCP_ACK are treated as a layer-2 data frames. Although the payload of TCP_ACK is small, transmission of TCP_ACK can consume a considerable amount of bandwidth due to the header and other overheads.

In our experiments, we consider the setup shown in Fig. 8. An FTP server is connected to the AP directly through an Ethernet. The FTP client is on a wireless station. So, in the AP buffer, VoIP packets intermix with TCP_DATA packets. At the same time, TCP_ACK packets sent from the FTP client will contend with TCP_DATA and VoIP packets sent from the AP, as well as with the VoIP uplink packets sent from all the VoIP clients.

We have also considered a file upload situation in which TCP_DATA is sent from the client to the server, and in which the TCP_ACK intermix with VoIP packets in the AP. The results will not be presented here since they are similar to those of the file download scenario presented here.

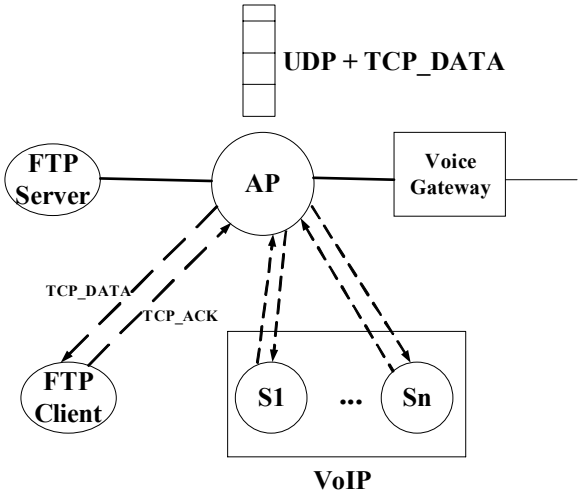

Figure 8. Setup for Experimental Studies of VoIP-TCP Interference

Table 9 shows the VoIP QoS metrics when six VoIP sessions coexist with one TCP connection. The TCP packet size is 1460 bytes. Here we only set up six VoIP sessions so that we can leave about half of the total WLAN bandwidth to TCP. The data shown here are those of one particular VoIP session. We have verified that other sessions have similar performance.

Table 9. Performance of Oridnary VoIP when six VoIP Sessions coexists with One TCP Connection

Access delay / jitter of the AP (ms)	Access delay / jitter of the station (ms)	VoIP downlink packet loss rate	VoIP uplink packet loss rate	TCP throughput (Mbps)
83.9 / 15.6	2.3 / 3.0	1.0 %	0	2.55

As can be seen, the voice quality is unacceptable even when there is only one TCP interference connection. The result can be explained as follows. The nature of TCP is such that after a connection is established, it will continue to increase the data input rate until packet losses occur. At the AP, this means TCP_DATA will continue to flood the buffer until the buffer overflows and packet losses occur. After that, TCP will decrease its input rate. Upon not having packet losses for a while, however, it will increase its input rate until the AP queue builds up again. The relatively high level of the buffer occupancy and the oscillatory data input rate of TCP leads to the high delay and jitter performance for the downlink VoIP stream observed in Table 9.

Solutions

A natural solution to the problem is Priority Queuing (PQ), in which voice packets are given priority over the TCP packets within the AP buffer. By limiting the number of VoIP sessions to below the VoIP capacity identified previously, TCP should be able to pick up the remaining WLAN bandwidth, and the use of PQ should not adversely affect

TCP throughput. That is, the performance gain for VoIP is not at the expense of TCP throughput.

Table 10 shows the delay and loss performance for VoIP when PQ is implemented in the AP buffer. Compared with Table 9, we see that PQ can drastically reduce the delay, jitter and packet loss rate of downlink VoIP packets. Furthermore, the TCP throughput suffers no degradation.

Table 10. Performance of Ordinary VoIP when six VoIP Sessions coexist with One TCP Connection with Priority Queuing at the AP

Access delay / jitter of the AP (ms)	Access delay / jitter of the station (ms)	VoIP downlink packet loss rate	VoIP uplink packet loss rate	TCP throughput (Mbps)
3.0 / 1.5	2.6 / 2.2	0.01 %	0	2.55

6.2 M-M VoIP coexisting with TCP over WLAN

In Section 4, we have shown that in a *pure* VoIP environment with no interfering TCP traffic, the lack of ARQ causes multicast packets in the M-M scheme to experience a high packet loss rate, especially when the voice sources are VBR. It turns out that the loss rate for the multicast VoIP packets can also be excessively when there is interfering uplink TCP traffic (with respect to Fig. 8, the interfering uplink TCP_ACK) even when the voice sources are CBR rather than VBR. This can be seen from the results in the first row of Table 11, in which six VoIP sessions in the M-M scheme coexist with one TCP connection.

Tables 9 and 11 both assume six VoIP sessions. With the M-M scheme in Table 11, however, the TCP throughput is higher. This is because the downlink VoIP packets are multiplexed into fewer multicast packets, leaving more bandwidth to TCP.

It can also be seen from the first row of Table 11 that not only is the loss rate of VoIP packets at the AP high, the delay is also unacceptable. The second row of Table 11 shows what happens when PQ is applied at the AP. Although the delay problem is solved, the loss rate remains excessively high. This is because the packet losses are caused by collisions with uplink unicast packets, not buffer overflow. Giving priority to multicast packets in scheduling transmissions of packets within the AP does not help to reduce these collisions. To reduce collisions, we must give priority to the AP multicast packets over unicast packets from other nodes. This requires us to look into the CSMA/CA scheme of 802.11 to find a solution.

In particular, we are interested in solutions that do not require changes to the 802.11 protocol used at the client stations. It turns out that a minor modification of the protocol used at the AP will do. We refer to the solution as the MAC-layer Multicast Priority scheme (MMP). With MMP, when the AP has a multicast frame to transmit, instead of waiting for DIFS and then a contention backoff period, it just waits for a Multicast Inter-Frame Space (MIFS), before transmission. The contention backoff period is omitted altogether. The value of MIFS should be a value less than DIFS but larger than SIFS. By setting it larger than SIFS, it will not collide with control frames such as ACK and CTS. By making it smaller than DIFS and getting rid of the contention backoff period,

collisions with unicast uplink packets are eliminated. In our simulation experiment, we set MIFS to be 30 us. Note that MMP is a general solution to the multicast collision problem in WLAN. That is, it is not limited to just VoIP multicasting. The restriction is that there should be no more than one multicast node within the WLAN; otherwise, multicast packets from different nodes would still collide. However, we believe in most multicast applications in an infrastructure-mode WLAN, the AP is likely to be the node from which multicast data is delivered to the clients.

Table 11. Performance of M-M when six VoIP Sessions coexist with one TCP Connection, with various Enhancement Schemes

	Access delay / jitter of the AP (ms)	Access delay / jitter of the station (ms)	VoIP downlink loss rate	VoIP uplink loss rate	TCP throughput (Mbps)
M-M	42.7 / 19.2	4.5 / 6.2	10.8 %	0	3.46
M-M + PQ	4.3 / 2.4	4.7 / 6.2	12.2 %	0	3.49
M-M + MMP	17.2 / 14.5	4.4 / 5.2	0	0	3.47
M-M + PQ+MMP	2.7 / 2.1	4.6 / 5.8	0	0	3.47

The third and fourth rows of Table 11 show what happens when MMP is applied, with and without PQ, respectively. As can be seen, VoIP packet loss at the AP has been eliminated altogether. Without PQ, the delay is still large; with PQ working in conjunction with MMP, both delay and loss become acceptable again.

Table 12. Performance of M-M when 11 M-M VoIP coexist with one TCP Connection, with Various Enhanacement Schemes

	Access delay / jitter of the AP (ms)	Access delay / jitter of the station (ms)	VoIP downlink loss rate	VoIP uplink loss rate	TCP throughput (Mbps)
M-M	32.5 / 25.8	6.6 / 10.2	15.6 %	0	2.55
M-M + PQ	4.5 / 3.2	6.7 / 13.5	12.0 %	0	2.54
M-M + MMP	20.3 / 21.7	8.9 / 20.8	0.2 %	0	2.54
M-M + PQ+MMP	2.9 / 2.7	5.8 / 7.2	0	0	2.54

Table 12 shows the results when the number of VoIP sessions is 11, half of the capacity of the M-M scheme when there is only VoIP traffic. Compared with Table 11, it is clear that TCP just picks up the remaining bandwidth in the WLAN after the VoIP traffic gets their share.

Recall that in Section 4, the capacity of the M-M scheme with VBR voice sources found from simulation was 36, far below the 50 derived from analysis. The fact that the channel was not fully utilized was due to the high collision rate suffered by multicast VoIP packets. The loss rate of uplink unicast VoIP packet was actually quite low. Since the MMP scheme removes the collisions of multicast packets, the VBR system capacity should in principle be improved. We have verified that this is in fact the case, and that the system capacity can be improved to 46 with good loss and delay performance. Although this is still below 50, it is reasonable, since the analysis was based on the average traffic load so that the capacity derived is at most an upper bound of the actual capacity.

To conclude this section, we would like to point out that giving priority to multicast traffic as in MMP will not cause significantly poorer performance for other MAC frames, since the multiplexed traffic load is relatively small. In addition, all the proposed solutions are AP-centric, and no changes to the client node's MAC layer are required. From the practical standpoint, the solutions can be more easily deployed, since the end users can use the current commercial products without any changes.

7. Further Discussions

We have assumed that there are no transmission errors in the WLAN in the preceding sections. In this section, we discuss our own experimental results regarding this assumption. In addition, we investigate the performance of 802.11e, specifically the EDCA mode of 802.11e, relative to our proposed scheme.

7.1 Transmission Errors

Our proposed MMP scheme can avoid collisions for multicast frame. However, it can not solve the reliability problem if multicast packets are lost due to transmission errors. Therefore, the packet loss characteristics due to transmission errors in a real environment are of interest to us. We have conducted several sets of real network experiments in our lab which have physical obstacles, microwave interferences and multi-path effects that may cause transmission errors. We believe the results obtained are representative of those in a typical office/lab environment where WLANs are deployed.

In our experiments, multicast packets of 500 bytes were transmitted from an AP to a wireless station. In the sender, we added a sequence number on every packet sent. Then the receiver located on the wireless station can calculate the packet loss rate based on the sequence number information. We measured the packet loss rates for various AP-station distances and data transmission rates. In particular, we use Lucent Orinoco AP to transmit multicast frame at 2 Mbps, Linksys AP to transmit multicast frame at 11 Mbps. The results are shown in Table 13.

Table 13. Multicast Packet Loss Rate for Different Distances and Data Rates

Distance (m)	Multicast frames at 2 Mbps	Multicast frames at 11 Mbps
1	0	0.17%
5	0	0.15%
10	0	0.17%
20	0.02%	0.23%

It can be seen that 11 Mbps data rate does lead to a higher packet loss rate than 2 Mbps. But within a reasonable distance (i.e., 20 meters, a typical range for an office or a lab), the multicast packet loss rate is negligible for both 2 Mbps and 11 Mbps data rates (relative to target 1% loss rate for VoIP applications). Our assumption of no transmission errors in the previous discussions is reasonable in that light.

7.2 802.11e

The IEEE 802.11 Working Group is currently defining a new supplement called 802.11e to the existing legacy 802.11 MAC sub-layer in order to support QoS. There are two access mechanisms in 802.11e, EDCA and HCCA, corresponding to the DCF and

PCF in the legacy 802.11 protocol. Since the focus of the preceding sections is DCF, here we will only study the corresponding part in 802.11e, EDCA.

Unlink DCF, which has one queue for all the traffic within one station, EDCA implements multiple queues within one station to provide differentiated, prioritized channel accesses for frames with different priorities. Each frame arriving at the MAC from the higher layers carries a specific priority value. Each priority is mapped into an access category (AC). Each AC has its own queue, and contends for the medium using a separate CSMA/CA instance.

ACs with different priorities are assigned different access parameters, such as inter-frame space called Arbitration Inter-Frame Space[AC] (AIFS[AC]), CWmin[AC], CWmax[AC]. The AP can adapt these parameters dynamically to the network conditions. Basically, the smaller AIFS[AC] and CWmin[AC], the shorter the channel access delay, and hence the more capacity allocated to the given traffic class. Collisions between ACs within the same station are resolved by granting access to the AC with the highest priority.

It is obvious that EDCA can not solve the low VoIP capacity problem because it does not reduce the protocol overhead. Since it has different queues for different types of traffic, it can be used to solve the unacceptable QoS problem when voice traffic coexists with data traffic.

Specifically, we can assign voice traffic a higher priority over data traffic. However, there is still the problem of how to properly set the access parameters in EDCA. When voice traffic coexists with data traffic, as the voice traffic load increases (i.e., the number of VoIP sessions increase), we should give data traffic less bandwidth so that to QoS of voice can be guaranteed. To illustrate the problem, we define the following parameter settings.

- EDCA0: One queue for all the traffic, same parameter setting as in DCF
- EDCA1: CWmin[voice] = CWmin[data] = 31
- EDCA2: CWmin[voice] = 31, CWmin[data] = 63
- EDCA3: CWmin[voice] = 31, CWmin[data] = 127

To simplify the problem, we set AIFS = DIFS, CWmax = 1031 for all the settings, as in DCF. We only change the CWmin for different priorities. This simplification will not affect our conclusions.

Table 14 and Table 15 show the performance of different EDCA parameter settings when one VoIP session coexists with one TCP, and when six VoIP sessions coexists with one TCP, respectively. In the former case, the EDCA1 setting is good enough to provide acceptable QoS for voice. EDCA2 and EDCA3 cause wastes of WLAN bandwidth. In the latter case, however, EDCA2 is optimal setting among the four. EDCA1 is under-tuned and EDCA3 is over-tuned. So when the number of VoIP sessions changes over time (as well as when the traffic load of other traffic changes), how to adaptively tune the parameter settings so that the limited WLAN bandwidth can be used efficiently is an outstanding problem for EDCA.

One the other hand, PQ + DCF as we mentioned previously does not require any parameter tunings. No matter how the number of VoIP sessions changes, PQ + DCF can always guarantee the QoS of voice while letting TCP take on the remaining bandwidth. That is, so long as the VoIP capacity is not exceeded, things will be fine.

Table 14. Performance of Different Parameter Settings for One VoIP + One TCP

	Access delay / jitter of the AP (ms)	Access delay / jitter of the station (ms)	TCP throughput (Mbps)
EDCA0	23.26 / 15.46	1.98 / 1.47	3.45
EDCA1	2.72 / 2.12	2.84 / 2.06	3.45
EDCA2	2.21 / 1.54	2.23 / 1.41	3.07
EDCA3	1.99 / 1.15	1.94 / 1.16	2.43

Table 15. Performance of Different Parameter Settings for Six VoIP + One TCP

	Access delay / jitter of the AP (ms)	Access delay / jitter of the station (ms)	TCP throughput (Mbps)
EDCA0	56.15 / 26.62	4.12 / 2.65	2.19
EDCA1	14.58 / 6.43	4.89 / 4.17	2.44
EDCA2	10.82 / 3.02	4.29 / 2.94	2.16
EDCA3	9.23 / 2.03	3.86 / 2.56	1.71

8. Conclusions

This paper investigates two critical technical problems in VoIP over WLAN: 1) low VoIP capacity in WLAN; 2) unacceptable VoIP performance in the presence of coexisting traffic from other applications. In setting out to find solutions to these two problems, 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. A salient feature of all the proposed schemes in this paper is that the MAC protocol at the wireless end stations needs not be modified, making them more readily deployable over the existing network infrastructure.

With regard to 1), we show that a Multiplex-Multicast (M-M) scheme can improve the VoIP capacity by close to 100%. The M-M scheme multiplexes the downlink VoIP packets at the AP into a larger multicast packet to reduce WLAN overheads. Unlike other VoIP capacity improvement schemes reported in the literature, the M-M scheme requires no changes to the standard 802.11 MAC protocol. Our studies are comprehensive and include various voice codecs, CBR and VBR VoIP streams, and 802.11b, 802.11a, and 802.11g MAC protocols. The results show that our proposed scheme can achieve a voice capacity 80% to 90% higher than ordinary VoIP in all cases, while meeting our performance target.

With regard to 2), our study shows that for both ordinary VoIP and M-M VoIP, the performance is unacceptable when there is co-existing TCP traffic in the WLAN. Two complementary schemes have been proposed and their effectiveness in solving the performance problem when used together has been demonstrated. The solutions only require some minor modifications at the AP.

This paper also considered the use of service differentiation mechanisms EDCA proposed in the 802.11e standard. The use of EDCA can not solve the problem 1) (low capacity problem) because it does not reduce the protocol overhead. It can be used to solve the problem 2) (QoS problem when voice coexists with data). But how to tune the parameters is still an outstanding issue to be investigated.

References

[1] Yi-Bing Lin and Imrich Chlamtac, *Wireless and Mobile Network Architectures*, John Wiley and Sons, 2001.

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

[3] D. Chen, S. Garg, M. Kappes, and K. Trivedi, "Supporting VBR VoIP Traffic in IEEE 802.11 WLAN in PCF Mode," *Technical Report ALR-2002-026*, *Avaya Labs*, 2002.

[4] M. Veeraraghavan, N. Cocker and T. Moors, "Support of Voice Services in IEEE 802.11 Wireless LANs," *INFOCOM 2001*, Vol. 1, Apr. 2001, pp. 488-497.

[5] Rusty O. Bladwin, Nathaniel J. Davis IV, Scott F. Midkiff and Richard A. Raines, "Packetized Voice transmission using RT-MAC, a wireless real-time medium access control protocol", *Mobile Computing and Communications Review,* Vol 5, No.3, pp. 11-25, July, 2001.

[6] S. Garg and M. Kappes, "On the Throughput of 802.11b Networks for VoIP," *Technical Report ALR-2002-012*, *Avaya Labs*, 2002. http://www.research.avayalabs.com/techreportY.html

[7] S. Garg and M. Kappes, "An Experimental Study of Throughput for UDP and VoIP Traffic in IEEE 802.11b Networks," *IEEE WCNC 2003*, Vol. 3, March 2003, pp. 1748-1753.

[8] T. Hiraguri, T. Ichikawa, M. Iizuka, and M. Morikura, "Novel Multiple Access Protocol for Voice over IP in Wireless LAN," *IEEE International Symposium on Computers and Communications*, July 2002.

[9] A. Banchs, X. Perez, M. Radimirsch, H. Stuttgen, "Service Differentiation Extensions for Elastic and Real-Time Traffic in 802.11 Wireless LAN," *IEEE Workshop on High Performance Switching and Routing*, May 2001, pp. 245-249.

[10] J. Kuri and S. K. Kasera, "Reliable Multicast in Multi-access Wireless LANs," *INFOCOM'99*, Vol. 2, Mar. 1999, pp. 760-767.

[11] Min-Te Sun, Lifei Huang, Arora, A. and Ten-Hwang Lai, "Reliable MAC layer multicast in IEEE 802.11 wireless networks," *Parallel Processing, Intl. Conf. on,* Aug. 2002, pp. 527-536.

[12] Tang, K. and Gerla, M., "MAC Layer Broadcast Support in 802.11 Wireless Networks," *MILCOM 2000*, vol. 1, Oct. 2000, pp. 544-548.

[13] 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.

[14] Hung-Huang Liu and Jean-Lien C. Wu, "Packet telephony support for the IEEE 802.11 wireless LAN", *IEEE Communications Letters*, Vol. 4, No. 9, pp. 286-288, Sept. 2000.

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

[16] Neeli Prasad and Anand Prasad, *WLAN Systems and Wireless IP for Next Generation Communications*, Artech House, 2002.

[17] A. Prasad, "Performance Comparison of Voice over IEEE 802.11 Scheme," *IEEE VTC 1999*, Vol. 5, Sept. 1999, pp. 2636-2640

[18] S. Casner and V. Jacobson, "Compressing IP/UDP/RTP headers for low-speed serial links", IETF RFC 2508, 1999.

[19] P. Brady, "A Model for Generating On-Off Speech Patterns in Two-Way Conversation," Bell Syst. Tech. Journal, vol. 48, no. 7, pp. 2245-2272, Sept. 1969.

[20] "The Network Simulator – ns2", http://www.isi.edu/nsnam/ns

[21] Floyd, S. and Fall, K., "Promoting the use of end-to-end congestion control in the Internet," *Networking, IEEE/ACM Trans. on*, Vol. 7, Aug. 1999, pp. 458-472.

[22] Law, K. L. E., "The bandwidth guaranteed prioritized queuing and its implementations," *Proc. GLOBECOM'97,* vol. 3, Nov. 1997, pp. 1445-1449.

[23] M. Visser and M. El Zarki, "Voice and Data Transmission over an 802.11 Wireless Network," IEEE Symposium on "Wireless Merging onto the Information Superhighway," Vol. 2, Sept. 1995, pp. 648-652.

[24] J. Feigin, K. Pahlavan, and M. Ylianttile, "Hardware-Fitted Modeling and Simulation of VoIP over a Wireless LAN," *52nd IEEE Vehicular Technology Conference*, Vol. 3, Sep 2000, pp. 1431-1438.

[25] T. J. Kostas, M. S. Borella, I. Sidhu, g. M. Schuster, J. Grabiec and J. Mahler, "Real-time voice over packet-switched networks,", *IEEE Network*, vol. 12, No. 1, pp. 18-27, January/February 1998.

[26] 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.

[27] B. Crow, I. Widjaja, J. Kim, P. Sakai, "IEEE 802.11 Wireless Local Area Networks," *IEEE Communications Magazine*, Vol. 35, Issue 9, Sept. 1997, pp. 116-126.

[28] D. Chen, S. Garg, M. Kappes, and K. Trivedi, "Supporting VoIP Traffic in IEEE 802.11 WLAN with Enhanced Medium Access Control (MAC) for Quality of Service," *Technical Report ALR-2002-025*, *Avaya Labs*, 2002.