The Voice Capacity of WiFi for Best Effort and Prioritized Traffic

Nghia T. Dao[*†], Xun Wei[*], Robert A. Malaney[*†]

[*]National ICT Australia (NICTA), Australian Technology Park, Eveleigh, NSW 1430, Australia.
[†]The School of Electrical Engineering and Telecommunications at the University of New South Wales, Sydney, NSW 2052, Australia.
email {dao.trong-nghia, xun.wei, robert.malaney}@nicta.com.au

***Abstract*—Crucial to supporting voice over 802.11b is the knowledge of voice capacity (N_c) of a single Access Point. This paper provides an analytical formulation of N_c in the case where all traffic in the network is voice. Our formulation, which can apply to a range of voice codec specifications, are verified by detailed simulations. We also investigate how to deliver, within the 802.11b standard, priority service to voice in the presence of best effort background traffic. It is known that the voice capacity degrades very quickly in the presence of other traffic sources if all packets are treated as best effort. Using an experimental deployment in which all voice packets are prioritized by having their channel back-off times set to zero, we determine the rate of the best effort background traffic below which our analytical formulation of voice-only capacity remains useful.**

***Index Terms*— WiFi, 802.11, MAC, VoIP.**

I. Introduction

WIRELESS local area networks have been widely and rapidly deployed, both in public spaces (airports, campuses, *etc*.,) and in the home networking environment. The most common protocol run over these networks is that based on the IEEE 802.11b standard [6], commonly known as WiFi. Services and applications running over WiFi are evolving from time-insensitive data applications to time-sensitive real-time applications. In particular, VoIP (Voice over IP) is anticipated to be widely deployed over WiFi in the next few years.

The VoIP capacity of a WiFi Access Point (AP), in terms of the number of simultaneous VoIP calls it can handle with acceptable Quality of Service (QoS), is an important parameter required for resource planning issues. In fact, this work is motivated by the desire to have a simple, but reasonably accurate, formulation of the VoIP capacity for a real-time QoS planning application known as QoS Seeker [3, 8]. QoS Seeker attempts to determine the dynamic QoS conditions in a WiFi network as a function of location. The VoIP capacity of an AP is critical to its functionality.

The current underlying technology of WiFi uses the 802.11b MAC protocol with DCF (Distributed Coordination Function) as the primary channel access mechanism. Previous studies of the VoIP capacity have been reported over a range of conditions and VoIP codecs [4, 5, 9]. These studies predict a range of N_c - the number of VoIP stations that can be accommodated by an AP at a minimum level of QoS. Typically, $4<N_c<30$. That this range for N_c is wide is a reflection of the different parameters which can impact the VoIP QoS and capacity. Such parameters include the codec rate, sampling period, data rates, traffic patterns, and wireless conditions. One of the aims of this work is to find a simple formulation for the value of N_c.

Another aspect of our work relates to the issue of priority services for VoIP over WiFi. In real-world deployment of WiFi networks it can be anticipated that other applications will be running over the network. It is well known that VoIP capacity can be seriously degraded in such a scenario [1]. In order to remedy this situation some manufacturers have developed APs which can prioritize VoIP traffic relative to other traffic types [12]. It is in fact possible to achieve such prioritization while still being within the 802.11b standard. Very little work has appeared in the literature related to priority VoIP services. It is another key aim of this work to identify the VoIP capacity of an AP when priority services are enabled.

As far as we are aware the work reported here is the first to consider analytical formulations, simulations and experiments in the one study of VoIP capacity. Also, previous works have focused on the wireless-to-wired VoIP capacity, whereas we focus on the capacity when all VoIP stations are in the wireless domain. Our formulation of voice capacity utilizes approximations not adopted in previous works, providing a surprisingly robust prediction of voice capacity. Finally, our experimental investigation of priority VoIP utilizes a mechanism for enabling priority services using a more simplified mechanism than that adopted in previous studies. The specific contributions of our work are threefold:

(i) We determine a simple formulation for voice capacity N_c, based on approximations for the timescale for VoIP packet transmission in WiFi and an estimation of the typical back-off period for all VoIP packets. OPNET simulations, and experiments, are carried out in order to verify our formulation.

(ii) We investigate, by a series of experiments, a specific mechanism to deliver priority services for VoIP over WiFi. The priority enabling mechanism we study is one based on setting the backoff parameter for all VoIP packets to zero.

(iii) We investigate the VoIP capacity of an AP under the assumption of priority VoIP services enabled via zero back-off. Our key aim is to determine the level of background traffic that our earlier (best effort) VoIP capacity formulation breaks down.

The rest of this paper is organized as follows. Section II provides the formulation for N_c. In section III we compare the analytical results with detailed simulations. Various experiments are carried out in section IV to investigate the performance of priority VoIP - enabled by setting (at the AP) the backoff parameter to zero for all voice traffic. Finally, concluding remarks are given in section V.

II. Analysis of WiFi voice capacity N_c

At the core of our attempts to formulate the VoIP capacity is the effective timescale for transmitting a VoIP packet in the WiFi protocol. This timescale is not simply the inverse of the data throughput (as might naively be expected), but a much larger timescale due to the protocol overhead, acknowledgements, and the collision avoidance mechanism. Figure 1 illustrates the process.

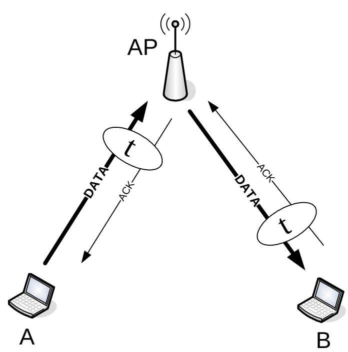

Figure 1. Sending a voice packet from A to B, via the AP

At any given time, the channel is utilized to transmit at most one frame. The transmitting frame from a station or AP can be a data frame or a control frame (RTS/CTS/ACK). Figure 1 shows the sequence of frames that need to be transmitted in the channel, in order to send one voice packet from A to B, via the AP. At A, the voice packet is encapsulated in a data frame and sent to the AP. Upon receiving the data frame without error, the AP generates an ACK to A, and forwards the frame to B. Finally, station B, after successfully receiving the frame, sends an ACK to the AP. It is clear that in order to send one voice packet from one wireless station to another, the wireless channel is busy for a total of four frame transmissions (two DATAs, and two ACKs). For simplicity, we assume that the stations experience good channel conditions and line-of-sight communications with each other. Under these conditions we are exploring the optimum voice capacity of the AP. Note, the RTS/CTS scheme is not required under these optimum conditions and is therefore not considered in our analysis. We will use the timescale t illustrated in Figure 1 directly in our estimate of N_c, *as* we now discuss.

The estimation of N_c can be based on a simple fact that the combined coding rate of all the VoIP stations must be less than the "effective throughput rate" of the wireless channel. This can be stated thus,

$$\frac{N_c C}{R_{eff}} \leq 1 \qquad (1)$$

where C is the voice coding rate at each station and R_{eff} is the effective throughput rate of the wireless channel. This can be written as $R_{eff} = \frac{b}{2t}$, where b is the voice packet size. From this we can approximate equation (1) as,

$$N_c \leq \frac{\rho}{2t} \qquad (2)$$

where ρ is the sampling period of the voice codec. In order determine a specific value of the bound it remains to specify the detailed form of the timescale *t.*

Figure 2 shows a breakdown of the timescale t required to send a data frame and receive an ACK.

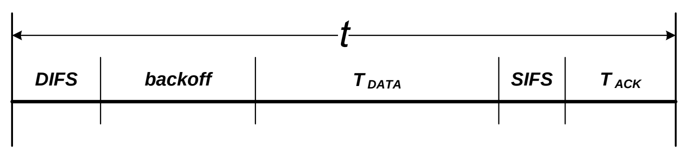

Figure 2. Time duration for successfully transmitting a data frame

In Figure 2, following the DCF basic access rule, a typical data frame needs to defer *DIFS* (Distributed Inter Frame Spacing) and wait an additional *backoff* time. After this time has elapsed, the data can be transmitted. Ignoring propagation delays, the timescale for this is T_{DATA}. The receiving station, after deferring *SIFS* (Short Inter Frame Spacing), will send an acknowledgement. Again, ignoring propagation delays, the timescale for this is T_{ACK}. As can be seen from Figure 2 the key timescale t can be written,

$$t = (DIFS + backoff + T_{DATA} + SIFS + T_{ACK}) \qquad (3)$$

Let us detail the individual values of the components in equation (3).

At the core of the collision avoidance mechanism in the WiFi protocol is the *backoff* parameter. This can be written as a function of the contention window (*CW*) variable,

$$backoff = rand[0, CW] * slot\_time \quad (4)$$

where the function *rand* uniformly samples the range 0 - *CW*, and *slot_time* is a time which is dependent on the physical technology (of order microseconds). At first, *CW* is set equal to CW_{min} = 31. To further reduce the probability of collision, after each unsuccessful transmission attempt, *CW* is doubled until a predefined maximum value CW_{max} = 1023 is reached. After each successful transmission, *CW* is reset to CW_{min}. The value of *backoff* is only counted down when the channel is sensed idle. When the *backoff* reaches zero, the station is allowed to transmit.

However, in this work we will adopt an approximation to the value of *backoff* which will simplify our VoIP capacity formulation. We use the fact that the number of retransmission attempts is very small when the number of stations approaches N_c (behavior observed in our simulations). Based on this observation, we omit any retransmission effects (*i.e.* the doubling of *CW* after collision detection) and consider the CW_{min} parameter only. Under the assumption that most voice packets are transmitted successfully in the first attempt, the average backoff time can be estimated as,

$$backoff = \frac{CW_{\min}}{2} Slot\_time \quad (5)$$

because in the first transmission attempt, the number of slots selected as backoff is uniformly chosen in range of [0, CW_{min}]. Only as the number of VoIP stations approaches N_c should the approximation implicit in equation (5) begin to falter.

The parameter T_{DATA} of equation (3) can be written:

$$T_{DATA} = T_{PLCP} + \frac{b+h}{L} \quad (6)$$

where T_{PLCP} is the physical layer overhead (in time units) which includes the PLCP preamble and PLCP header fields, *b* is the voice packet payload size, *h* is the total header size of the frame (including RTP, UDP, IP and MAC headers), and *L* is the physical data rate (in range $L_{min} \le L \le L_{max}$).

Finally, the T_{ACK} of equation (3) can be written as,

$$T_{ACK} = T_{PLCP} + \frac{a}{L} \quad (7)$$

where *a* is the acknowledgement frame size. Here we have assumed the highest data rate is used for transmitting the acknowledgement.

The necessary physical parameters required to give specific values to our terms for 802.11b DSSS (Direct Sequence Spread Spectrum) are given in Table I.

TABLE I
PHYSICAL PARAMETERS

	802.11b DSSS
Slot Time	20 µs
SIFS	10 µs
DIFS	50 µs
CW_{min}	31
CW_{max}	1023
L_{min}, L_{max}	1 Mbps, 11 Mbps
PLCP overhead	192 µs

Adopting these values and using equation (2), we get the expression for N_c as,

$$N_C = \left\lfloor \frac{500\rho}{774 + \frac{592 + C\rho}{L}} \right\rfloor \quad (8)$$

where the voice sampling period ρ is in units of ms, the voice coding rate *C* is in units of Kbps, and the physical data rate *L* is in units of Mbps. The form of brackets used indicates that the lowest integer value is to be used.

Note that equation (8) is based on the use of CBR (constant bit rate) voice traffic and not traditional on-off voice models. CBR traffic was adopted in [4] and [5], whereas [9] utilized a four-state Markov chain voice model, taking into account all possible states of the calling and called parties (ITU P.59 recommendation [13]). If traditional on-off models are used for voice traffic the value of N_c can be estimated by multiplying the right–hand-side of equation (8) by a factor of $1/\alpha$ where α is the voice activity cycle parameter (α = 0.42 is typical). This claim is substantiated later when we consider our simulations.

The analysis just given differs from work reported previously in several aspects. The analysis and experiments of [5], the analysis and simulations of [4, 9], and the experiments of [1] were all based on wireless to wired scenarios. Our formulations, however, were based on the wireless-only scenario (in which both ends of each voice call are in the wireless domain). This scenario is useful when WiFi is utilized as a cellular-like infrastructure. Secondly, in our analysis we utilized an approximation not used in [4, 5, 9]. This approximation is that the value of the *backoff* parameter is *always* set as shown in equation (5). This helps simplify our formulation considerably, leading to a simpler (dependent on fewer parameters) expression for the voice capacity. Even though or formulation of the voice capacity depends on only three parameters (sampling period, coding rate, and physical data rate) we shall see from our simulations that equation (8) still provides a robust estimate of voice capacity.

III. Simulations of WiFi voice capacity

It is clear that we have made numerous simplifying assumptions in arriving at our formulation of the VoIP capacity as given by equation (8). It is therefore important that we verify our analytical predictions via simulation. To this end we carried out numerous OPNET simulations [10] of VoIP over WiFi. In the simulations the VoIP stations are configured to talk to each other in pairs. This means, each station is talking with exactly one other station. Each simulation is run for 100 seconds.

In terms of QoS, the voice performance is considered acceptable if the end-to-end (E2E) delay is less than 150ms and the packet loss rate is less than 2%. Which one of these QoS metrics is most important will be dependent on the AP buffer size. In our simulations, the AP buffer size is set to 32K Bytes, and because of this we find that the packet loss rate is the factor that limits the voice capacity N_c. We can see this from the results of Figure 3 and Figure 4. Here the voice traffic is CBR with 64Kbps coding rate and 10ms, 20ms and 30ms sampling period. For the 10ms case (the left hand curve in each figure), it can be seen that when the number of stations is equal to six, the voice packet E2E delay is still in an acceptable range whereas the packet loss rate exceeds 10%. The main reason causing packet loss rate is buffer overflow at the AP. Similar results are observed with different coding rates and sampling periods.

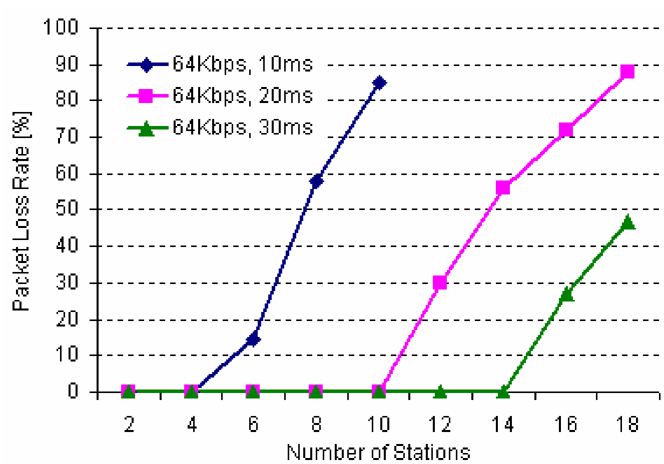

Figure 3. Packet Loss Rate vs. Number of Stations

Although it may appear that increasing the AP buffer size beyond 32K Bytes would reduce packet loss rate, and therefore increase N_c, this turns out not to be the case. Packets queued beyond the 32K Bytes limit would incur unacceptable delays. To verify this, we carried out simulations with 6 stations (CBR voice traffic with 64Kbps and 10ms sampling rate) and varying AP buffer sizes. We observed that the AP queue gets full when the number of stations is six, even when the buffer is increased from 16K Bytes to 64K Bytes. Although the packet loss rate reduces, the voice E2E delay increases (Figure 5), and becomes the primary factor that limits N_c. No change in the value of N_c is predicted for large buffer sizes.

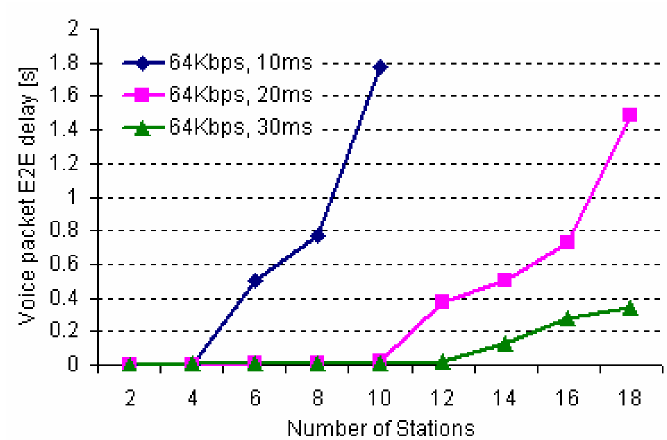

Figure 4. Voice packet E2E delay vs. Number of Stations

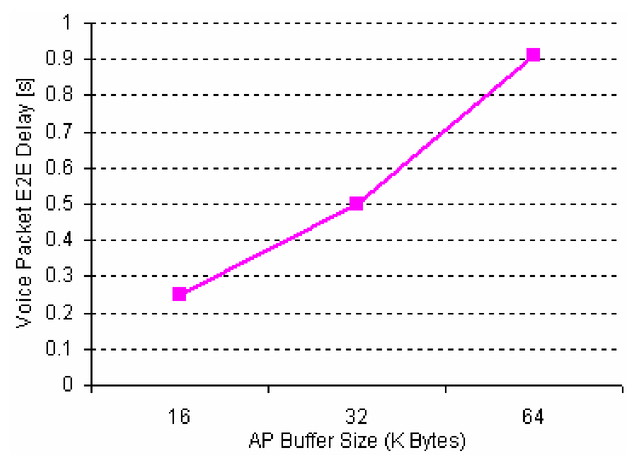

Figure 5. Voice packet E2E delay vs. AP Buffer Size

In Table II, the VoIP capacity determined via equation (8) for different values of the coding rate and sampling period are listed. Here we have assumed that 802.11b DSSS is operating at the maximum data rate, and that the voice traffic is CBR. The numbers in brackets are simulation results for the same configurations. It can be seen that in general there is good agreement between the analytical predictions and the simulation results.

TABLE II
Voice Capacity

	10 ms	20 ms	30 ms	50 ms	100 ms
64 kbps	5 (4)	10 (10)	14 (14)	22 (20)	35 (30)
32 kbps	5 (4)	11 (10)	16 (14)	25 (22)	44 (38)
8 kbps	6 (6)	11 (10)	17 (16)	28 (26)	55 (44)

We also carried out numerous simulations where we adopted on-off models for the voice traffic with a voice activity cycle parameter of α = 0.42. These simulations confirmed that a simple multiplication of the right-hand-side of equation (8) by the factor $1/\alpha$ provided good agreement between simulation and analytical formulation. Note that our tests of a popular VoIP application [11], placed its sampling period close to 30ms, showing that its expected voice capacity is in the range 14-17.

For further verification of the value N_c, we carried out experiments which probe configurations where capacity limits are small - the number of VoIP stations available to us limits our ability to probe high capacity configurations. Using a LinkSys WRT54G (v3.1) AP [14], we configured VoIP experiments in which 802.11b-enabled laptops were connected to each other in pairs. Each laptop was set to generate CBR traffic. This setup matched our simulation configurations. In a series of experiments we increased the number of VoIP stations by two, and measured packet round trip time and loss rate in each case. The average value for each experiment is given in Table III for the case where the sample period was 10ms.

TABLE III
EXPERIMENT RESULTS

Number of Stations	2	4	6	8
RTT [ms]	15	53	483	587
Loss Rate [%]	0	0.2	34	63

We can observe that, when the number of stations increases from 4 to 6, a sharp increase in both packet loss and E2E delay is measured. This verifies that N_c equals 4 when the voice sampling period is 10ms and the coding rate is 64kbps, a result in good agreement with that provided in Table II for this configuration.

IV. PRIORITY SERVICE FOR VOIP

In the previous section we have focused on the capacity of an AP in the case where there is no background (non-VoIP) traffic. However, in real-world deployment of WiFi networks it can be anticipated that other applications will be running over the network. It is well known that VoIP capacity can be seriously degraded in such a scenario. In order to remedy this situation some manufacturers have developed APs which can prioritize VoIP traffic relative to other traffic types. It is in fact possible to achieve such prioritization while still being within the 802.11b standard. One way of achieving this is to specify that the backoff parameter (at the AP) is set to zero whenever the packet to be transmitted is a VoIP packet [12]. An additional method of delivering priority service is to ensure that when packets are queuing at the AP for transmission, any prioritized VoIP packet is placed in the queue ahead of any other packet. Here we will focus just on the zero backoff method. This distinguishes our work from previous studies of priority VoIP capacity [10]. It should be straightforward to override the backoff parameter in most standard commercial APs, whereas additions of separate queuing mechanisms are more involved. Our principal aim is to determine at what level of background traffic our previous results on capacity analysis remain valid when zero backoff is applied to all VoIP packets.

In our priority-enabled experiments we used a Cisco Aironet 1231G access point [11]. The Cisco Aironet 1231G can prioritize packets by checking the IP precedence information in the IP header TOS field. In our experiments we define packets with an IP precedence value of 7 as VoIP packets, whereas an IP precedence value of 0 is assigned to all background packets. We give VoIP packets zero backoff time by setting both CW_{min} and CW_{max} to zero. For background traffic the value CW_{min} and CW_{max} is set to the standard 802.11b values. Background traffic was generated by 802.11b-enabled Windows XP stations. These stations ran Iperf [12], arranged to generate 2Mbps CBR UDP background traffic. The size of the background packets was set at 1500 Bytes. One Windows XP station was used as a sender of the background traffic, and a distinct Windows XP station was used as a receiver. A VoIP traffic generator was deployed on Pocket PCs running Windows CE 4.2. In the experiments we measured the round-trip delay between two VoIP stations, which we then halved to get the estimated one-way delay.

In our first experiment we show how important priority service is for VoIP when background traffic is present. The E2E delay for four VoIP stations (all running 64Kbps codecs and sampling periods 10ms) was measured. The values for a particular VoIP station (the other stations showed similar results) are shown as circles (blue) in Figure 6 for the case of 2Mbps background traffic, with normal backoff for VoIP. The adverse affect of the background traffic on the VoIP QoS can be clearly seen, with average E2E delays of 1.5 seconds measured. Compare this with the E2E delay of the VoIP packets when no background traffic is present - shown as the squares (purple) in Figure 6. This latter data, which effectively forms a straight line at the bottom of the plot, indicates that all four VoIP stations can be readily accommodated by the channel at approximately 30ms E2E delay. Clearly the presence of the background traffic is the major issue leading to zero VoIP capacity. The need clearly exists for some form of prioritized transmission behavior for the VoIP packets.

Figure 7 shows the results of applying zero backoff to all VoIP packets at the AP. Shown are tests for a constant sample rate of 10ms for different codec rates.

For each codec rate the experiment used the number of VoIP stations equal to the value of N_c. predicted by simulation.

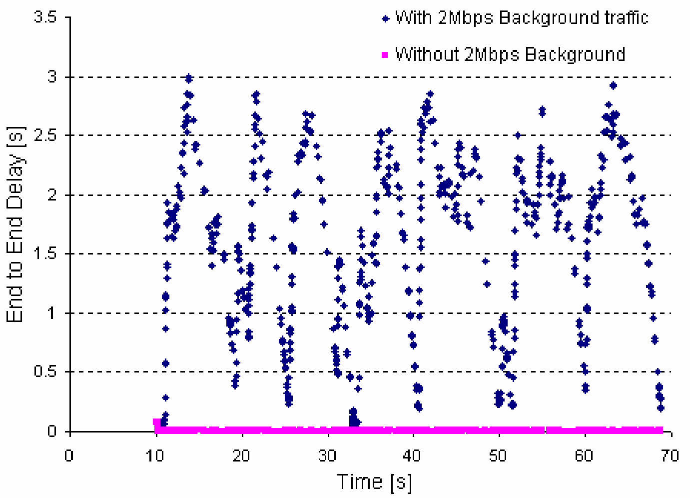

Figure 6. Voice packet E2E delay vs. Time

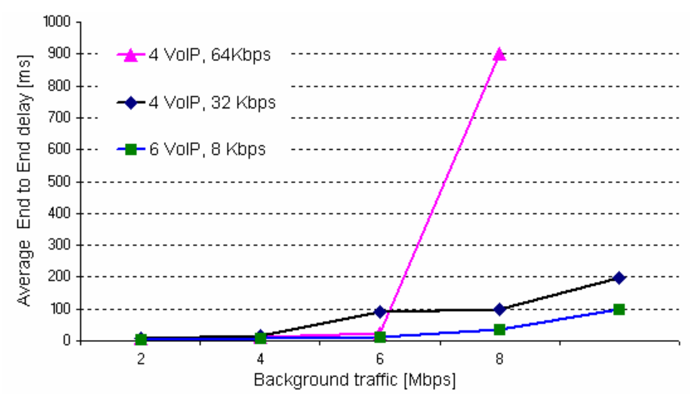

Figure 7. Voice packet E2E delay vs. AP Buffer Size

The average E2E delay is shown plotted as a function of the background traffic. What we can see here is that the average E2E delay becomes unacceptable at about 6 Mbps. This shows that below about 6Mbps the capacity analyses of the previous section can be used if zero backoff is enabled at the AP.

V. Conclusion

We have investigated the voice capacity, N_c, of a single AP. We have suggested a simple formulation of N_c based on the average timescale for VoIP packet transmission in WiFi. Our formulation is convenient for estimating the voice capacity, and can apply for any voice codec specifications. We carried out extensive simulations and experiments which show that our formulation is surprisingly robust. Also we showed that, without any priority scheme, the voice capacity N_c is drastically reduced, even with a small volume of background traffic. When the backoff time at the AP is set to zero for all voice traffic, a priority service for VoIP is enabled. In this priority scheme we show that the voice performance is remarkably improved. Moreover, we show that within the zero-backoff priority scheme, our formulation of N_c will remain valid if the volume of background traffic is below about 6 Mbps.

Acknowledgment

This work has been supported by funding from National ICT Australia (NICTA). The authors are also grateful Dr. Ravindra Ranasinghe for useful discussions.

References

[1] Anjum, F.; Elaoud, M.; Famolari, D.; Ghosh, A.; Vaidyanathan, R.; Dutta, A.; Agrawal, P.; Kodama, T.; Katsube, Y.; "Voice performance in WLAN networks - an experimental study", Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE
Volume 6, 1-5 Dec. 2003 Page(s):3504 - 3508 vol.6

[2] Cisco Systems, Inc.; "Cisco 1200 series access point QoS configuration guide"

[3] Dao, N.; Malaney, R.A.; Exposito, E.; Wei, X.; "Differential VoIP Service in WiFi Networks and Priority QoS Maps", Global Telecommunications Conference, 2005. GLOBECOM '05.

[4] Garg, S.; Kappes, M.; "Can I add a VoIP call?", Communications, 2003. ICC '03. IEEE International Conference on Volume 2, 11-15 May 2003 Page(s):779 - 783 vol.2

[5] Hole, D.P.; Tobagi, F.A.; "Capacity of an IEEE 802.11b WLAN supporting VoIP", Communications, 2004 IEEE International Conference on Volume 1, 20-24 June 2004 Page(s):196 - 201

[6] IEEE 802.11 WG, Reference number ISO/IEC 8802-11:1999(E) IEEE Std 802.11, 1999 edition. International Standard [for] Information Technology - Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific Requirements – Part 11:Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, 1999.

[7] Iperf; http://dast.nlanr.net/Projects/Iperf/

[8] Malaney, R.A.; Exposito, E.; Wei, X.; Dao, N.; "Seeking VoIP QoS in Physical Space", Proc. of the Third ACM International Workshop on Wireless Mobiles Applications and Services on WLAN Hotspots (WMASH), pp23 -32, Sept. 2005.

[9] Medepalli, K.; Gopalakrishnan, P.; Famolari, D.; Kodama, T.; "Voice capacity of IEEE 802.11b, 802.11a and 802.11g wireless LANs", Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE Volume 3, 29 Nov.-3 Dec. 2004 Page(s):1549 - 1553 Vol.3

[10] OPNET Technologies, Inc.; http://www.opnet.com

[11] Skype Technologies S.A.; http://www.skype.com

[12] SpectraLink Corporation.; http://www.spectralink.com/

[13] ITU Recommendation P.59, "Artificial Conversational Speech," March 1993.

[14] Cisco-LinkSys, LLC.; WRT54G Access Point.