Understanding 802.11e Voice Behaviour via Testbed Measurements and Modeling

Ian Dangerfield, David Malone, Douglas J. Leith

Abstract—**Analytic modeling and simulation of 802.11 and 802.11e have flagged up a number of potential performance problems with voice in an infrastructure mode network. This body of work has also predicted a number of possible solutions, such as the use of $TXOP$. In this paper we reproduce a number of these problems in a 802.11 testbed, and demonstrate that analytic models usefully bound performance. We study the impact of buffer size, which is difficult to fully capture in analytic models. We observe an asymmetry in buffer requirements between stations and the access point and explore how buffering and the use of $TXOP$ impact throughput and delay of voice.**

Keywords— **802.11e, CSMA/CA, buffering, test-bed, voice, measurement**

I. Introduction

The 802.11e MAC protocol extends the standard 802.11 CSMA/CA contention mechanism by allowing the adjustment of MAC parameters that were previously fixed. While the 802.11e protocol has been extensively studied in the literature, this work is almost entirely confined to analytical and simulation studies. Owing to the lack of available hardware, there have been few experimental studies evaluating the performance of the new 802.11e protocol. Hardware is, however, now available which allows us to investigate 802.11e EDCA operation in a real testing environment. We have constructed an 802.11e hardware testbed network and in this paper our aim is make use of this testbed to perform experimental measurement and validation of 802.11e operation.

As a first step, in [1] we compared our expectations (from theory and simulation) with the throughput behaviour of an actual 802.11e implementation. There is an extensive literature containing simulation [2], [3] and analytic [4], [5] studies/comparisons of the 802.11 and 802.11e MAC mechanisms. A number of experimental studies, albeit in the context of 802.11 rather than 802.11e, also show a gap between theoretical predictions and practical performance [6], [7]. In [8] we developed a technique for measuring MAC level delays in our 802.11e testbed and verified the use of 802.11e to protect a single delay sensitive flow from saturated data flows.

In [1] we also considered the impact of 802.11 causing the access point (AP) to be a bottle neck for data traffic. This is also known to be a limiting factor for the number of voice calls a network can support [9], [10]. Modelling and simulation has suggested that by prioritising the AP voice capacity can be increased. In this paper we implement these schemes in a test bed and compare the experimental

measurements with model predictions, where possible. We also look at the impact of varying the buffer space available, an effect that is difficult to capture in analytic modeling, but has received some simulation based study [11]. Some modeling work has also confirmed that increased buffering alone is not a solution to voice unfairness [12].

In this paper the 802.11e parameters CW_{min} and $TXOP$ will be of interest. CW_{min} is the minimum contention window and is adjustable in powers of two i.e. as 2^k with integer k. $TXOP$ how long a station may transmit when it wins a transmission opportunity. It is a length of time, specified in microseconds.

II. Testbed Setup

The 802.11e wireless testbed is configured in infrastructure mode. It consists of a desktop PC acting as an access point, 18 PC-based embedded Linux boxes based on the Soekris net4801 [13] and one desktop PC acting as client stations. The PC acting as a client records delay measurements for each of its packets, but otherwise behaves as an ordinary client station. All systems are equipped with an Atheros 802.11b/g PCI card with an external antenna. The system hardware configuration is summarised in Table I. All nodes, including the AP, use a Linux 2.6.8.1 kernel and a version of the MADWiFi [14] wireless driver modified to allow us to adjust the 802.11e CW_{min}, and $TXOP$ parameters. All of the systems are also equipped with a 100Mbps wired Ethernet port, which is used for control of the testbed from a PC. Specific vendor features on the wireless card, such as turbo mode, are disabled. All of the tests are performed using the 802.11b physical maximal transmission rate of 11Mbps with RTS/CTS disabled and the channel number explicitly set. Since the wireless stations are based on low power embedded systems, we have tested these wireless stations to confirm that the hardware performance (especially the CPU) is not a bottleneck for wireless transmissions at the 11Mbps PHY rate used. As noted above, a desktop PC is used as a client to record the per-packet delay measurements. This is to ensure that there is ample disk space, RAM and CPU resources available so that collection of statistics not impact on the transmission of packets.

The configuration of the various network buffers and MAC parameters is detailed in Table II. As we will be interested in the impact of buffering, we have varied the length of certain queues.

The testbed was calibrated as described in [1] by adjusting the positions of stations and antennae until the throughputs achieved by all stations were roughly similar.

Hardware	model	spec
1× AP	Dell GX 260	2.66Ghz P4
16× node	Soekris net4801	266Mhz 586
1× measurement node	Dell GX 270	2.8Ghz P4
WLAN NIC	D-Link DWL-G520	Atheros AR5212

TABLE I

Testbed Hardware Summary

parameter	default	used
total buffering	399 packets	b chosen from 3,4,15,30,399
interface tx queue	199 packets	$b/2 - 1$ packets
driver tx queue	200 packets	$b/2 + 1$ packets
MAC Preamble	short	long
MAC Data rate	54Mbps	11Mbps
MAC ACK rate	2Mbps	11Mbps
MAC Retries	11	11

TABLE II

Testbed Parameters Summary

Several software tools are used within the testbed to generate network traffic and collect performance measurements. To generate voice-like network traffic we use a custom program. While many different network monitoring programs and wireless sniffers exist, no single tool provides all of the functionality required and so we have used a number of common tools including tcpdump[15]. Network management and control of traffic sources is carried out using ssh over the wired network.

A key feature that distinguishes voice traffic from data traffic is its sensitivity to network delay. In the present context we are interested, in particular, in the network delay associated with winning access to transmission opportunities in an 802.11 WLAN. This MAC access delay is associated with the contention mechanism used in 802.11 WLANs. The MAC layer delay, i.e. the delay from a packet becoming eligible for transmission (reaching the head of the hardware interface queue) to final successful transmission, can range from a few hundred microseconds to hundreds of milliseconds, depending on network conditions.

The measurement technique that is used is to only use the clock on the sender, to avoid the need for synchronisation. By requesting an interrupt after each successful transmission we can determine the time that the ACK has been received. We may also record the time that the packet was added to the hardware queue, and by inverting the standard FIFO queueing recursion we can determine the time the MAC spent processing the packet. This process is illustrated in Figure 1. We have refined the technique described in [8] by using hardware provided timestamps and can achieve measurements accurate to a few microseconds.

III. Impact of Buffering on Voice Capacity

Consider an 802.11b WLAN carrying n two-way voice conversations. That is, we have n wireless stations (STAs) each transmitting the voice of one speaker and n replies
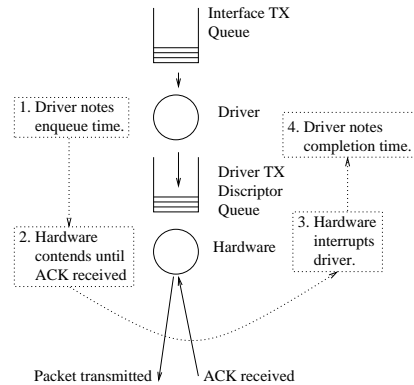

Fig. 1. Schematic of delay measurement technique.

transmitted by the AP[1].

Since voice calls are two-way, it is the throttling at the AP that currently limits voice call capacity in infrastructure networks rather than the overall wireless bandwidth available. Specifically, with the 802.11 MAC each station has approximately the same number of transmission opportunities. This includes not only the wireless stations, but also the AP itself. Suppose we have n_u wireless stations each transmitting upstream traffic and n_d downstream flows transmitted by the AP. When the network is saturated we expect that the n_u wireless stations have roughly a $n_u/(n_u + 1)$ share of the bandwidth while the AP has only a $1/(n_u + 1)$ share. As the network comes near saturation, this asymmetry that can result in the AP becoming the network bottleneck.

We now study this network using the testbed described in Section II. Unless specifically noted otherwise, results shown are experimental results from this setup.

Figure 2 shows the throughput at the AP when the AP and the STAs all have the standard 802.11b MAC parameters. We consider buffer sizes of 3, 4 and 30 packets. The buffer size at the AP and the STAs is set to be the same[2]. We also show the predictions of two analytic models [17], [12] which predict the throughput using a 1 packet buffer and an infinite buffer[3]. It can be seen that the model predictions accurately bound the measurements.

Note that the AP's throughput is substantially improved by using a buffer size of 30, rather than a small buffer of 3 or 4 packets. However, even with a buffer of 30 packets there is a consistent drop in throughput above about 12 stations. This is in contrast to the stations, whose throughput is shown in Figure 3, including model based predictions. Here results are noisier because of the lack of multiplexing, however this lack of multiplexing also leads to a smaller buffer requirement because each station is only serving a

[1]Parameters for the voice calls are taken from [16]: 64kbs on-off traffic streams where the on and off periods are exponentially distributed with mean 1.5 seconds, but periods of less than 240ms are increased to 240ms in length, to reproduce the minimum talk-spurt. Traffic is two-way; the on period of an upstream call corresponds to the off period of its downstream reply.

[2]Each configuration is run for at least 20 minutes.

[3]Model parameters (transmission and collision duration) were obtained by testbed delay measurement.
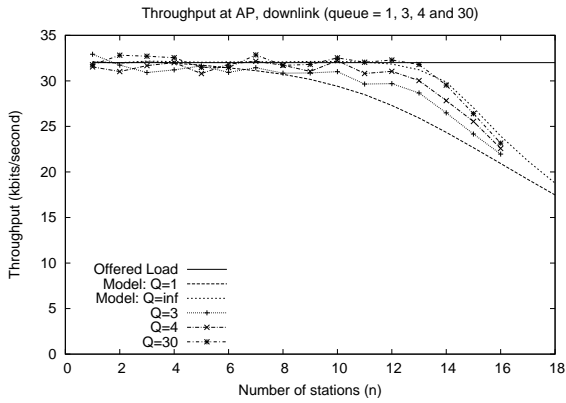
Fig. 2. Measured and modelled throughput at the AP for standard 802.11b MAC settings. Results are shown for various buffer sizes. The buffer size at the AP and the STAs is the same. Predictions using analytic modeling for a buffer size of 1 and an infinite buffer are shown and bound achieved throughput.
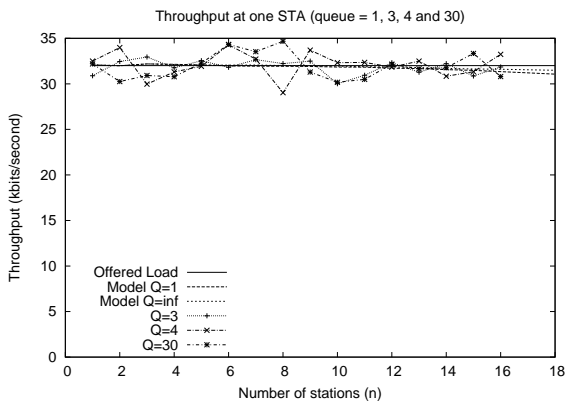

Fig. 3. Measured and modelled throughput at a single STA for standard 802.11b MAC settings. The buffer size at the AP and the STAs is the same. Results are shown for various buffer sizes.
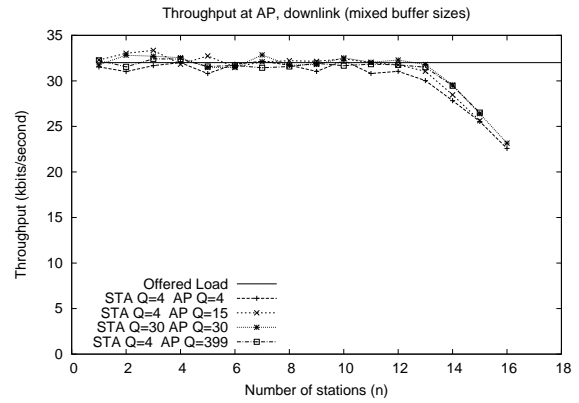

Fig. 4. Measured throughput at the AP for standard 802.11b MAC settings. Results are shown for various combinations of buffer sizes at the STAs and the AP.
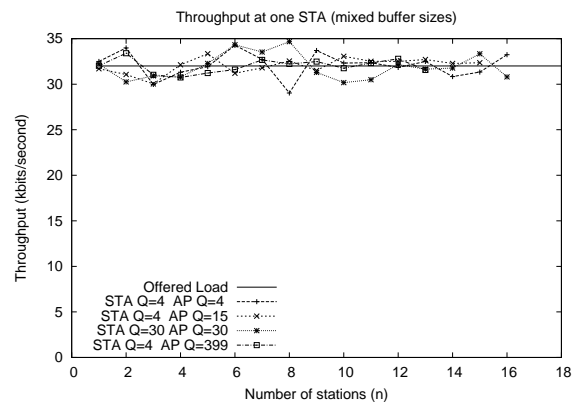

Fig. 5. Measured throughput at a single STA for standard 802.11b MAC settings. Results are shown for various combinations of buffer sizes at the STAs and the AP.

single source and does not have to deal with short-term buffering of packets from multiple sources.

To demonstrate that there is an asymmetry in buffering requirements, Figure 4 shows resulting AP throughput where the buffering at the AP and the stations is not the same. Figure 5 shows the corresponding STA throughput. From this we see that though buffer size affects throughput at the AP, STA throughput is insensitive to buffer size, a buffer of 4 packets is essentially sufficient at the stations.

Figure 4 also shows that the performance at the AP is quite similar for 15, 30 and 399 packet buffers, in contrast to a 4 packet buffer where throughput drops by almost 10% at 10 nodes. This is in broad agreement with the predictions of the infinite buffer model shown in Figure 2. We believe this is because the main purpose of buffering at the AP is accommodate the multiplexing of sources, and that once we have buffering proportional to n at the AP, there is little advantage to adding more. In the remainder of the paper, for simplicity, we will set the buffer at the STA and the AP to be the same. However we expect the STA's requirements to be much smaller.

Voice is sensitive to both throughput and delay performance. Figure 6 shows the cumulative distributions for MAC delay at the access point for a buffer size of 4 (too small for the AP) and 30 (OK for the AP). We show the delay distribution for three different numbers of STAs: 5 (underloaded), 10 (close to saturation) and 15 (overloaded). The delay shows some basic features such as an initial step, representing a minimum transmission time, and slight kinks, representing numbers of backoff stages/other STA's transmissions before the final successful transmission.

As we expect, in the underloaded situation the AP buffer size makes little difference to the delay distribution. Even for 10 stations, increasing buffering has little impact, which is surprising as increased buffering has a notable impact on throughput (cf. Figure 2). When overloaded, buffering has a small impact, slightly increasing the delay. We believe this is because when the network is busy, extra buffering allows more packets to reach the MAC layer for transmission. Each packet must contend for access and this reduces delay performance.

Figure 7 shows the corresponding delay distribution at a station. While increasing buffering only had a small impact on delay for the AP, the increase is slightly larger for the
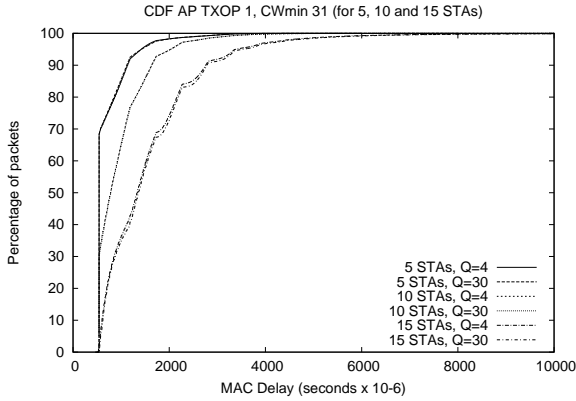
Fig. 6. Cumulative distribution for delays at the access point when there are 5, 10 and 15 stations with standard 802.11b MAC settings.
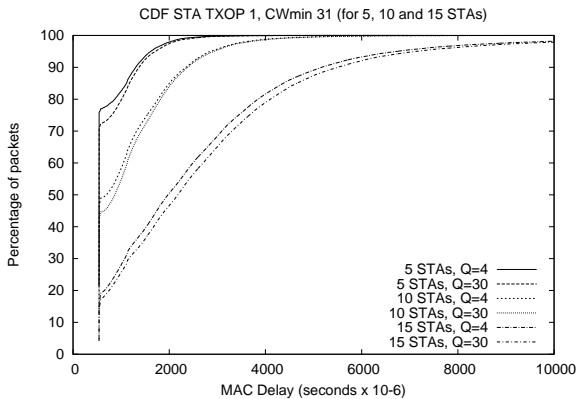

Fig. 7. Cumulative distribution for delays at a station when there are 5, 10 and 15 stations with standard 802.11b MAC settings.

Fig. 8. Throughput at the AP for prioritised voice, with $TXOP = n$ packets. Results are shown for various experimental and modelled buffer sizes.

Fig. 9. Throughput at the AP for prioritised voice, with $TXOP = n/2$ packets, $CW_{min} = 15$. Results are shown for various experimental and modelled buffer sizes.

STA, resulting in a shift of a few tenths of a millisecond. This may be due to a combination of factors, for example we know increased buffering results in increased throughput at the AP and a consequent reduction in free time on the medium should result in increased transmission times for stations.

IV. COMBINED IMPACT OF PRIORITISATION AND BUFFERING

By restoring parity in transmission opportunities between forward and reverse traffic in infrastructure networks the potential exists to increase voice call capacity. We consider prioritising the AP to restore parity between upstream and downstream flows. In the most simple case, we use the $TXOP$ packet bursting mechanism to increase the number of packets that the AP can send at each transmission opportunity to be n_d, the number of downstream voice flows. We also consider the scheme where $TXOP$ allows the AP to send $n_d/2$ frames, but we also half the value of CW_{min} to approximately double the number of transmit chances available to the AP. The aim of this is to see if frequent but smaller bursts have a more favourable impact on the delays of stations in the system.
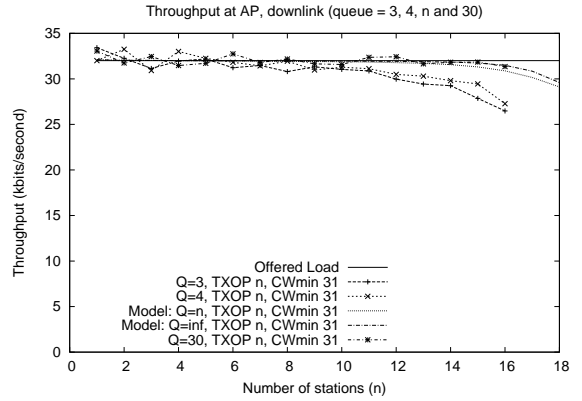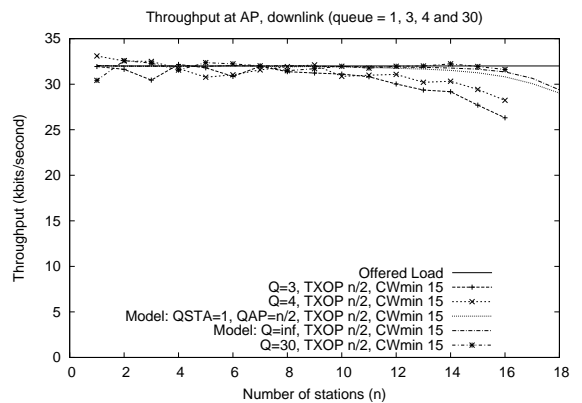
Figure 8 and Figure 9 respectively show the throughput using these prioritised schemes. We see that 3 and 4 packet buffers are still insufficient, but for 30 packet buffers we can maintain throughput well above 90% of average load of 32Kbps out as far as 16 voice calls in the network. The models we use can model an infinite buffer or a buffer of n packets. We see that these models predict almost the same throughput at a level consistent with our 30 packet buffer measurements. The throughput performance of the two schemes is similar. Intuitively we might expect this, as giving the AP a transmission of half the size twice as frequently should be similar as long as there is sufficient buffering to accommodate the bursts.

Regardless of buffering or prioritisation at the AP, throughput at our measurement station remained satisfactory for the range of STAs that we explored, as shown in Figure 10 (the other scheme shows similar performance). As shown, this is consistent with the predictions of the analytic models.

Figure 11 shows the mean MAC delay of packets at the access point. As the MAC delay corresponds to the service time of the queue, we require the mean service time to be less than the mean packet inter-arrival time. As expected, we see that prioritising the AP decreases the mean delay.
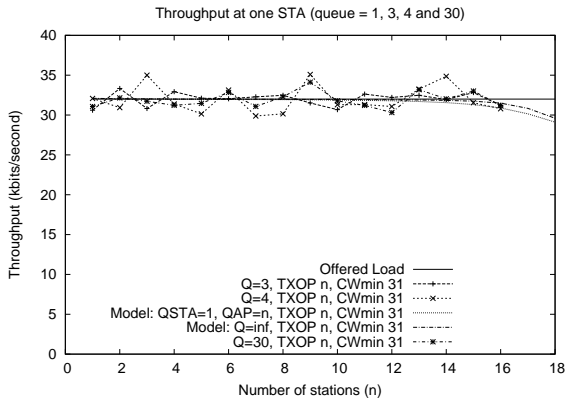
Fig. 10. Throughput at a STA for prioritised voice, with $TXOP = n$ packets at the AP. Results are shown for various buffer sizes.
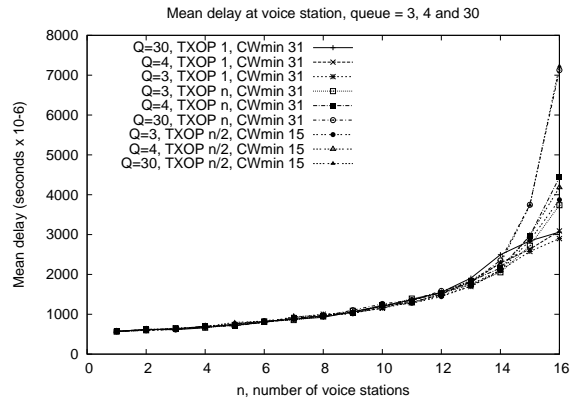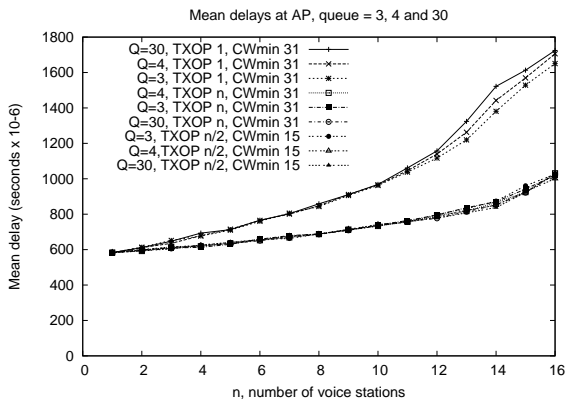

Fig. 11. Mean MAC delay at the AP for standard 802.11b MAC settings and when the access point is prioritised using 802.11e. Results are shown for various buffer sizes. The group with longer mean delays correspond to the experiments in which the AP is not prioritised.

Fig. 12. Mean MAC delay a station for standard 802.11b MAC settings and when the access point is prioritised using 802.11e. The mean inter-packet arrival time at a STA is $10000\mu s$.
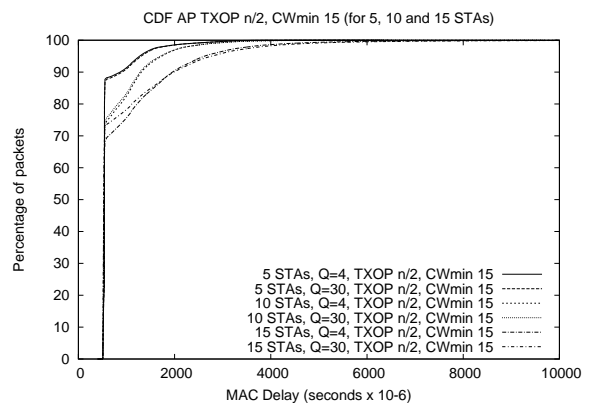

Fig. 13. Cumulative distribution for delays at the access point when there are 5, 10 and 15 stations when the AP is prioritised using $TXOP = n/2$ packets and $CW_{min} = 15$.

We can see that the impact of the queueing is largely insignificant. There is a small impact: without prioritisation we see that increasing the queue size increases the MAC delay, and so *decreases* the service rate. This is because increased buffering is leading to increased contention. Conversely, increasing the queue size in the prioritised case actually slightly *lowers* the mean service time. We believe this is because it increases the sizes of the bursts that the access point sends and so increases efficency without increasing contention.

Figure 12 shows that the stations' delay increases more rapidly when the AP is prioritised and the the number of stations is more than 14. Examining the performance for 16 stations reveals three groups: those where the AP is unprioritised (≈ 3000ms delay), those where the AP is prioritised but has a small buffer (≈ 4000ms delay) and those where the AP is prioritised with a large buffer (≈ 7000ms delay). We have seen that to maintain AP throughput we need both a larger buffer and prioritisation, so we conclude that we are nearing the boundary of the stability region for STAs. The exact prioritisation used does not have a significant impact on the stations' mean delay. This is inter-

esting, as one might expect longer bursts to lead to longer mean delays.

Figure 13 shows the delay cumulative distributions at the access point when using $TXOP = n/2$ packets and $CW_{min} = 15$. The results for $TXOP = n$ are similar. Compare Figure 6 and Figure 13. Naturally, prioritising the AP has improved the delay distribution in all cases.

However, the impact of buffering is now slightly different to the unprioritised case (Figure 6). Again, when the system is below saturation we see buffering having little impact. However, when the system is overloaded, in the prioritised situation increased buffering actually makes the delay distribution marginally better. This may be due to increased buffering allowing longer $TXOP$ bursts, and this can increase the rate at which packets are cleared without significantly increasing the amount of contention on the network.

The use of $TXOP$ at the AP may also have an impact on the stations. A large burst of packets transmitted by the AP might have a detrimental effect on the jitter experienced by a station's packet. As bursts are smaller in the prioritised scheme with $TXOP = n/2$ packets and
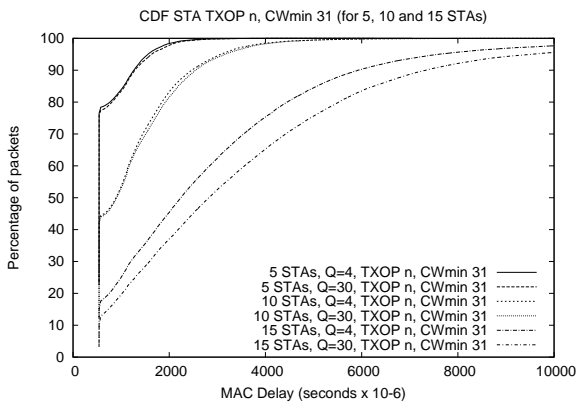
Fig. 14. Cumulative distribution for delays at a station when there are 5, 10 and 15 stations when the AP is prioritised using $TXOP = n$ packets.
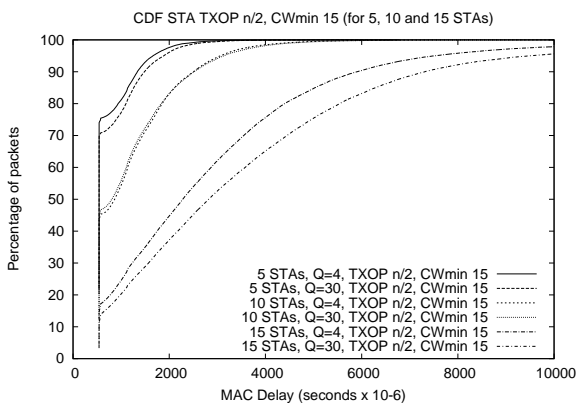

Fig. 15. Cumulative distribution for delays at a station when there are 5, 10 and 15 stations when the AP is prioritised using $TXOP = n/2$ packets and $CW_{min} = 15$.

$CW_{min} = 15$, we might expect it to be more kind to the STA. Figure 14 shows the delay distribution at a STA when the access point has long bursts given by $TXOP = n$ packets. Compared to Figure 7 there is little change, perhaps a marginal worsening of the delay distribution, however it seems to be limited to a few tenths of a millisecond in below-saturation cases. Figure 15 shows the results when the AP is prioritised with smaller, but more frequent bursts. While there again are some small differences, for the underloaded and below-saturation examples there are no substantial differences between the scheme.

Note that increased buffering in the system uniformly results in no improvement the STA's delay distribution, regardless of the prioritisation or load in the system. This is in contrast to the situation at the AP where the impact of buffering depended on the presence of prioritisation. This is in line with our observations that little buffering is sufficient for an individual STA, and increased buffering at the AP may hinder a STA's performance.

V. Conclusion

We have demonstrated that throughput at the AP is the main constraint on capacity for voice traffic in infrastructure mode 802.11 networks, (e.g., reducing the capacity to less than 10 calls), confirming previous analytic and simulation based studies. We show that buffering can partially alleviate this problem (e.g., allowing up to 12 calls). We note that the buffering requirements at the access point are different from those at a station; while a station requires approximately fixed buffering the access point requires buffering proportional to the number of calls. $TXOP$ and CW_{min} are shown to be effective in further resolving this problem (e.g., increasing the capacity to around 16 calls). The impact on delay and jitter introduced by $TXOP$ does not seem to pose a barrier to its use. We have also confirmed a subtle interaction between buffering and contention which can be observed by measuring MAC service times, which may indicate a small advantage to using smaller buffers for delay sensitive traffic.

References

[1] A. Ng, D. Malone, and D. Leith, "Experimental evaluation of TCP performance and fairness in an 802.11e test-bed," in *ACM SIGCOMM Workshops*, 2005.

[2] Q. Ni, "ns-802.11e EDCF for IEEE 802.11e wireless LAN," http://www-sop.inria.fr/planete/qni/, INRIA, Tech. Rep., November 2002.

[3] S. Wiethölter and C. Hoene, "Design and verification of an IEEE 802.11e EDCF simulation model in ns-2.26," Technische Universität Berlin, Tech. Rep. TKN-03-019, November 2003.

[4] G. Bianchi, "Performance analysis of IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, March 2000.

[5] R. Battiti and B. Li, "Supporting service differentiation with enhancements of the IEEE 802.11 MAC protocol: models and analysis," University of Trento, Tech. Rep. DIT-03-024, May 2003.

[6] E. Pelletta and H. Velayos, "Performance measurements of the saturation throughput in IEEE 802.11 access points," in *WiOpt*, April 2005.

[7] M. Franceschinis, M. Mellia, M. Meo, and M. Munafo, "Measuring TCP over WiFi: A real case," in *WiNMee*, April 2005.

[8] I. Dangerfield, D. Malone, and D. Leith, "Experimental evaluation of 802.11e EDCA for enhanced voice over WLAN performance," in *International Workshop On Wireless Network Measurement (WiNMee)*, 2006.

[9] N. Hegde, A. Proutiere, and J. Roberts, "Evaluating the voice capacity of 802.11 WLAN under distributed control," in *Proc. IEEE LANMAN*, September 2005.

[10] P. Clifford, K. Duffy, D. Leith, and D. Malone, "On improving voice capaicty in 802.11 infrastructure networks," in *Proc. IEEE WirelessCom*, June 2005.

[11] D. Malone, P. Clifford, and D. Leith, "On buffer sizing for voice in 802.11 WLANs," *IEEE Communications Letters*, vol. 10, no. 10, October 2006.

[12] K. Duffy and A. Ganesh, "Modeling the impact of buffering on 802.11," *IEEE Communications Letters*.

[13] "Soekris engineering," http://www.soekris.com/.

[14] "Multiband Atheros driver for WiFi (MADWiFi)," http://sourceforge.net/projects/madwifi/, r1645 version.

[15] V. Jacobson, C. Leres, and S. McCanne, "tcpdump," http://www.tcpdump.org/.

[16] A. Markopoulou, F. Tobagi, and M. Karam, "Assessing the quality of voice communications over internet backbones," *IEEE Transactions on Networking*, vol. 11, no. 5, pp. 747–760, October 2003.

[17] K. Duffy, D. Malone, and D. Leith, "Modeling the 802.11 Distributed Coordination Function in non-saturated conditions," *IEEE Communications Letters*, vol. 9, no. 8, pp. 715–717, 2005.