On Improving Voice Capacity in 802.11 Infrastructure Networks

Peter Clifford, Ken Duffy, Douglas J. Leith and David Malone
Hamilton Institute, NUI Maynooth,
Co. Kildare, Ireland
Email: {peter.clifford, ken.duffy, doug.leith, david.malone}@nuim.ie

Abstract—In this paper we consider voice calls in 802.11e infrastructure mode networks. We demonstrate that in standard 802.11 WLANs it is throttling of voice traffic at the access point that is the primary limitation on voice call capacity. We demonstrate that a straightforward 802.11e prioritisation strategy avoids this throttling and yields close to the theoretical maximum call capacity.

I. INTRODUCTION

In recent years, 802.11 wireless LANs have become pervasive. While providing wire-free connectivity at low cost, it is widely recognized that the 802.11 MAC layer requires greater flexibility and the new 802.11e standard consequently allows tuning of MAC parameters that have previously been constant. Although the 802.11e standard provides adjustable parameters within the MAC layer, the challenge is to use this flexibility to achieve enhanced network performance.

In this paper we study the behavior of infrastructure mode 802.11 networks where traffic is transmitted via an access point (AP). Our starting point is the observation that the 802.11 Carrier Sensing Multiple Access/Collision Avoidance (CSMA/CA) mechanism enforces per station fairness, i.e. each station has approximately the same number of transmission opportunities. This includes not only the wireless stations (STAs), but also the AP itself. We show that this has profound implications for network performance.

Consider an 802.11b WLAN carrying n two-way voice conversations. There are n wireless stations each transmitting the voice of one speaker and n replies transmitted by the AP1. Figure 1 shows throughput, loss and delay for both the AP and the aggregate of the wireless stations, with increasing number of voice conversations. As the number of conversations rises above 8, the throughput achieved by the AP falls relative to that of the wireless stations. When the number of calls exceeds approximately 10, the loss-rate of the downstream AP traffic increases beyond a viable level. These results are with small buffers, so queueing delays are short and loss is the limiting factor.

We compare this behavior with the following simple capacity calculation. Table I gives the overhead budget for the

transmission of a small packet of payload 80 bytes. With overhead the transmission of an 80 bytes payload takes 651µs, thus the maximum possible user throughput is approximately 80B/651µs = 0.98Mb/s. That is, for 64kb/s voice conversations, the channel can at best support 0.98/0.64 = 15.4 calls, which is 50% higher than the above measured capacity. The figure of 15.4 calls is over-optimistic as it neglects the idle time spent during contention window count-down as well as many other details of the channel behavior (such as packet collisions). Nevertheless, it suggests that room for improvement may well exist and it is this which is the subject of the present paper.

As voice calls are two-way, it is evident from Figure 1 that the throttling action at the AP currently acts as a primary limiting factor in voice call capacity in infrastructure networks (rather than the overall wireless bandwidth available – we
can see that the MAC delay remains low, indicating that the channel remains relatively lightly loaded with collisions and associated exponential backoff of the 802.11 contention windows infrequent). By restoring parity between forward and reverse traffic in infrastructure networks the potential exists to increase voice call capacity. In this paper we investigate how we can use the flexibility provided by the new 802.11e MAC to avoid AP throttling in infrastructure WLANs and thereby increase network capacity.

<table>
<thead>
<tr>
<th>Slot time, $\sigma$</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Propagation delay, $\delta$</td>
<td>1</td>
</tr>
<tr>
<td>$CW_{\text{min}} = 32 \sigma$</td>
<td>640</td>
</tr>
<tr>
<td>DIFS (AIFS=0)</td>
<td>50</td>
</tr>
<tr>
<td>SIFS (Short Inter Frame Space)</td>
<td>10</td>
</tr>
<tr>
<td>PLCP Header @1Mb/s</td>
<td>192</td>
</tr>
<tr>
<td>MAC Header 24 Bytes @1Mb/s</td>
<td>17.5</td>
</tr>
<tr>
<td>CRC Header 4 Bytes @1Mb/s</td>
<td>2.9</td>
</tr>
<tr>
<td>IP Header 20 Bytes @1Mb/s</td>
<td>14.5</td>
</tr>
<tr>
<td>MAC ACK 14 Bytes @1Mb/s</td>
<td>112</td>
</tr>
<tr>
<td>$E_i$ payload 80 Bytes @1Mb/s</td>
<td>58.2</td>
</tr>
<tr>
<td>$E_i$ payload 540 Bytes @1Mb/s</td>
<td>392.73</td>
</tr>
</tbody>
</table>

### TABLE I
802.11b MAC VALUES, BASIC RATE 1Mbps AND DATA RATE 11Mbps.

## II. RELATED WORK

There have been a number of previous studies of voice over 802.11 networks. Most have been concerned with measuring the voice call capacity of 802.11 networks rather than adjusting the MAC layer behavior itself. For example, in [2] a back-of-the-envelope calculation for maximum capacity of a WLAN is presented and shown to be a useful estimate. The authors also consider, using simulation, how delay constraints and bit-error rates impact the capacity of the network. Other metrics for voice capacity are used in [3] and [4].

## III. 802.11 CSMA/CA

The 802.11 MAC layer uses a CSMA/CA algorithm with binary exponential back-off to regulate access to the shared wireless channel. Briefly, on detecting the wireless medium to be idle for a period DIFS, each station initializes a counter to a random number selected uniformly from the interval [0,CW-1]. Time is slotted and this counter is decremented for each slot that the medium is idle. An important feature is that the count-down halts when the medium becomes busy and only resumes after the medium is idle again for a period DIFS. On the counter reaching zero, the station is permitted to transmit for a time $TXOP$ on the medium (defined to be one packet in 802.11). If a collision occurs (two or more stations transmit simultaneously), CW is doubled and the process repeated. On a successful transmission, CW is reset to the value $CW_{\text{min}}$ and a new count-down starts for the next packet. The new 802.11e MAC enables the values of $DIFS$ (called $AIFS$ in 802.11e), $CW_{\text{min}}$ and $TXOP$ to be set on a per class basis for each station, with a maximum of four classes per station.
packets are destined. This provides a direct measure of the number \( n_d \) of active downstream flows in a manner which is both straightforward and dynamically adapts to accommodate bursty and intermittent traffic. The effectiveness of this scheme is shown in Figure 2, where it can be seen to restore fairness between the upstream and downstream flows.

**Comment: Packet Burst Size.** With this TXOP approach the AP transmits \( n_d \) packets in a single burst. For \( n_d \) large, this can result in the AP occupying the channel for a substantial consolidated period of time and this may, for example, negatively impact competing delay sensitive traffic. We can address this issue in a straightforward manner by using multiple smaller bursts instead of a single burst. When using smaller packet bursts, it is necessary to ensure a corresponding increase in the number of transmission opportunities won by the AP. This can be achieved by using a smaller value of \( CW_{\text{min}} \) for the downstream traffic class at the AP. It is shown in [5] that competing traffic classes gain transmission opportunities approximately in inverse proportion to their values of \( CW_{\text{min}} \). Let \( k \) denote the ratio of the stations upstream class \( CW_{\text{min}} \) value to that of the downstream class at the AP. Scaling \( k \) with the number of transmission opportunities required provides coarse (recall that in 802.11e \( k \) is constrained to be a power of two) prioritization of downstream flows. We then complement this with use of TXOP for fine grained adjustment of the packet burst lengths, scaling TXOP with \( 1/k \). Hence fine grained prioritization can be achieved while avoiding unduly large packet bursts.

**V. Voice Calls**

While the foregoing argument provides insight and makes accurate predictions for greedy traffic flows, the situation with voice calls is more complex. We can see this immediately from Figure 1 where the upstream and downstream voice flows achieve almost equal throughput up to around 8 calls. In contrast, if the upstream/downstream flows were greedy (always have a packet to send) then the foregoing analysis indicates that with 8 calls the upstream flows would in aggregate achieve a factor of 8 greater throughput than the downstream flows.

We can understand this behavior by noting that, firstly, voice traffic is relatively low rate and so need not make use of every available transmission opportunity awarded by the 802.11 MAC. Secondly, a voice conversation involves speakers approximately taking turns at talking. That is, traffic is between pairs of speakers with the on period of one speaker roughly corresponding to the off period of the other. Both of these features mitigate the contention between the wireless stations and the AP for access to the wireless channel.

To explore this behavior further, in the next section we discuss a heterogenous finite-load analytic model suited to voice traffic modeling. We then use this, together with simulations, in Section V-B to study the impact of AP prioritization on the behavior of voice calls in infrastructure mode networks.

**A. Analytic Modeling**

Following the seminal paper of Bianchi [6], much of the analytic work on 802.11 MAC performance has focused on saturated networks where each station always has a packet to send. In particular, recent work has extended the saturation modeling approach to include multi-class 802.11e networks, see [5], [7] and [8]. The saturation assumption is key to these models as it enables queueing dynamics to be neglected and avoids the need for detailed modeling of traffic characteristics, making these networks particularly tractable.

Networks do not typically operate in saturated conditions. Internet applications, such as voice over IP, video and web browsing all exhibit bursty or on-off traffic characteristics. Creating an analytic model that includes fine detail of traffic arrivals and queueing behavior, as well as 802.11 MAC operation, presents a significant challenge. We introduce an 802.11 model with traffic and buffering assumptions that make it sufficiently simple to give explicit expressions for quantities of interest (throughput per station and collision probabilities), but still capture key effects of non-saturated operation. Although our traffic assumptions form only a subset of the possible arrival processes, we will see they are useful in modeling a wide range of traffic, in particular voice conversations.

Details of our analytic model are contained in the Appendix; using specified 802.11 MAC parameters and arrival rates, the model predicts transmission probability, collision probability and throughput. The predictive accuracy of the model is illustrated in Figures 4 and 5, where model predictions are compared with throughput data from NS packet-level simulations, with 540 Byte payloads, as the arrival rate is varied across its range and as the number of wireless nodes is varied. The collision probabilities corresponding to Figure 4 are shown in Figure 3 (similar accuracy is obtained for the conditions used in Figure 5).

The utility of the model for modeling VoIP traffic, both in peer-to-peer and infrastructure mode networks, is demonstrated in Figure 1. It can be seen that the model makes remarkably accurate predictions.

**B. Voice Traffic**

As before, we consider setting the AP TXOP to be equal to the number of active downlink voice calls. Figure 6 shows throughput and delay with this scheme as the number of

![Fig. 3. Collision probability as the traffic arrival rate is varied for smaller numbers of nodes (Model and NS simulation, 802.11 MAC parameters in Table I).](image-url)
voice calls is increased. It can be seen that now both the AP and wireless stations achieve similar throughputs, with the simulation and model results in good agreement. The network can now sustain approximately 15 voice calls before the delay exceeds the packet inter-arrival time of 10 ms and the system enters an unstable queueing regime where delays and loss rapidly increase.

The capacity is also in good agreement with the simple calculation given in the Introduction which establishes that the best case capacity is 15.4 voice calls. We can see that the measured capacity of 15 calls is remarkably close to this ideal value\(^2\) which indicates that the scope for further performance improvement is limited.

\(^2\)With \(CW_{min} = 32\), we expect that the average count-down time is \(CW_{min}/2 = 16\) slots or \(320\mu s\). Using this value the capacity of the wireless channel falls to only 10 calls using our simple calculation. A measured capacity of 15 voice calls implies that the average count-down time is significantly less than \(CW_{min}/2\). This can be explained by noting that when voice packets are generated by many stations statistical multiplexing of packet transmissions takes place. Hence, the average time the medium is idle is much less than the average per flow backoff time.

VI. CONCLUSIONS

In this paper our objective has been to develop a soundly-based strategy for selecting 802.11e MAC parameters in infrastructure networks carrying voice traffic. Specific contributions include the following:

- A demonstration that gross unfairness can exist between upstream and downstream flows in 802.11 infrastructure networks. This is associated with MAC level enforcement of per station fairness.
- The introduction of an new analytic model of 802.11 networks is developed that is capable of capturing the behavior of voice traffic.
- A straightforward 802.11e prioritization strategy for infrastructure networks that avoids throttling of voice flows at the AP.

The proposed 802.11e prioritization approach has the merit of being very straightforward and of imposing only a small computational burden (e.g. we use simple parameter settings and complex measurement-based adaptation is avoided). The approach is compatible with the WME subset of 802.11e that is supported by currently available hardware.

VII. ACKNOWLEDGMENTS

This work was supported by Science Foundation Ireland grant 03/I1N3/1396.

APPENDIX

Bianchi [6] presents a Markov model where each station is modeled by a pair of integers \((i, k)\). The back-off stage,
the state, naturally, 1 extra successful increase in the station is enabled quantitatively: the station is transmits and so the station is idle during a typical slot, the transitions from the (0,0) state are:

\[
P[(0,0)_e|(0,0)_e] = 1 - q + \frac{qP_{idle}(1-p)}{W_0},
\]

\[
k > 0, \quad P[(0,k)_e|(0,0)_e] = \frac{qP_{idle}(1-p)}{W_0},
\]

\[
k \geq 0, \quad P[(1,k)|(0,0)_e] = \frac{qP_{idle}p}{W_0},
\]

\[
k \geq 0, \quad P[(0,k)|(0,0)_e] = \frac{q(1-p_{idle})}{W_0}.
\]

Observe that \(p\), the probability of a collision given that we are about to transmit, is the probability that at least one other station is transmitting. This is also the probability that the medium is busy if we know the station under consideration has been silent. Hence we substitute \(P_{idle} = 1 - p\).

Given the collision probability \(p\) for this station in the system and per-station parameters \(q\), \(W_i\) and \(m\) we may solve for a stationary distribution of this Markov chain. This will enable us to determine the probability \(\tau\) that this station is attempting transmission in a typical slot.

First we make observations that aid in the determination of the stationary distribution. With \(b(i, k)\) and \(b(0, k)_e\) denoting the stationary probability of being in states \((i, k)\) and \((0, k)_e\), as \(b\) is a probability distribution we have

\[
\sum_{i=0}^{m} \sum_{k=0}^{W_i-1} b(i, k) + \sum_{k=0}^{W_0-1} b(0, k)_e = 1. \quad (1)
\]

We will write all probabilities in term of \(b(0, 0)_e\) and use the normalization in equation (1) to determine \(b(0,0)_e\). We have the following relations. To be in the sub-chain \((1, k)\), the following must have occurred: a collision from state \((0, 0)_e\) or an arrival to state \((0,0)_e\) followed by detection of an idle medium and then a collision, so that \(b(1,0) = b(0,0)_e + b(0, 0)_e q(1-p)p\). Neglecting packet discard, for \(i > 1\) we have \(b(i, 0) = p^i b(1,0)\) and so

\[
\sum_{i=1}^{\infty} b(i, 0) = b(1, 0) \frac{1-p}{1-p} = \frac{b(0, 0)_e p + b(0, 0)_e q(1-p)p}{1-p}. \quad (2)
\]

The keystone in the calculation is then the determination of \(b(0, W_0-1)_e\). Transitions into \((0, W_0-1)_e\) from \((0,0)_e\) occur if there is an arrival, the medium is sensed idle and no collision occurs. Transitions into \((0, W_0-1)_e\) also occur from \((i, 0)_e\) if no collision and no arrival occurs

\[
b(0, W_0-1)_e = b(0, 0)_e \frac{(1-p)(1-q)}{W_0} + \frac{(1-p)(1-q)}{W_0} \sum_{i \geq 0} b(i, 0) . \quad (3)
\]

Combining equations (2) and (3) gives

\[
b(0, W_0-1)_e = b(0, 0)_e \frac{(1-p)(1-q)}{W_0} + b(0, 0)_e \frac{1-q}{W_0}.
\]

We then have for \(W_0 - 1 > k > 0\), \(b(0, k)_e = (1-q) b(0, k+1)_e + b(0, W_0-1)_e\), with \(b(0, k)_e\) on the left hand side replaced by \(qb(0,0)_e\) if \(k = 0\). Straight forward recursion leads to
expressions for \( b(0, k)_e \) in terms of \( b(0,0)_e \) and \( b(0,0) \), and so we find

\[
\frac{b(0,0)}{b(0,0)} = \frac{q}{1-q} \left( \frac{1}{qW_0 - (1-p)(1-pq)W_0} \right) \tag{4}
\]

Using these equations we can determine the second sum in equation (1)

\[
W_0 - 1 \sum_{k=0}^{W_0 - 1} b(0, k)_e = b(0,0)_e \frac{qW_0}{1 - (1-q)W_0}.
\]

The \((0, k)\) chain can then be tackled, starting with the relation

\[
b(0, W_0 - 1) = \sum_{i \geq 0} b(i,0) \frac{(1-p)q}{W_0} + b(0,0)_e \frac{qp}{W_0}.
\]

Recursion leads to

\[
W_0 - 1 \sum_{k=0}^{W_0 - 1} b(0, k)_e = b(0,0)_e \left[ \frac{q}{1-q} \left( \frac{W_0 + 1}{2} \right) + \frac{q^2W_0 + qW_0 + q - 2}{2(1-q)W_0} + 1 - q \right].
\]

Using equation (4) we can determine \( b(1,0) \) in terms of \( b(0,0)_e \):

\[
b(1,0) = b(0,0)_e \frac{pq^2}{1-q} \left( \frac{W_0}{1 - (1-q)W_0} - (1-p)^2 \right).
\]

Finally, the normalization (1) gives

\[
1/b(0,0)_e = (1-q) + \frac{q^2W_0(W_0+1)}{2(1-q)W_0} + \frac{q(W_0+1)}{2(1-q)} \frac{q^2W_0}{W_0} + \frac{p(1-q) - q(1-p)^2}{2(1-q)(1-p)} \left( \frac{W_0}{1-(1-q)W_0} - (1-p)^2 \right) \tag{5}
\]

The main quantity of interest is \( \tau \), the probability that the station is attempting transmission. A station attempts transmission if it is in the state \((i,0)\) (for any \(i\)) or if it is in the state \((0,0)_e\), a packet arrives and the medium is sensed idle. Thus \( \tau = q(1-p)b(0,0)_e + \sum_{i \geq 0} b(i,0) \), which reduces to

\[
\tau = b(0,0)_e \left( \frac{q^2W_0}{(1-p)(1-q)(1-q)W_0} - \frac{q^2(1-p)}{1-q} \right), \tag{6}
\]

where \( b(0,0)_e \) is given in equation (5), so that \( \tau \) is expressed solely in terms of \( p, q, W_0 \) and \( m \). While \( q, W_0 \) and \( m \) are fixed for each station, in order to determine the collision probability, \( p \), we must give a relation between the stations competing for the medium.

Consider the case where \( n \) stations are present, labeled \( l = 1, \ldots, n \). Equation (6) gives an expression for \( \tau_l \), the per-station transmission probability, in terms of a per-station arrival process \( q_l \) and a per-station collision probability \( p_l \). Observe that

\[
1 - p_l = \prod_{j \neq l} (1 - \tau_j), \text{ for } l = 1, \ldots, n, \tag{7}
\]

that is, there is no collision for station \( l \) when all other stations are not transmitting. With \( n \) stations, (6) and (7) provide \( 2n \) coupled non-linear equations which can be solved numerically for \( p_i \) and \( \tau_i \).

The length of each state in the Markov chain is not a fixed period of real time. Each state may be occupied by a successful transmission, a collision or the medium being idle. To convert between states and real time, we must calculate the expected time spent per state, which is given by

\[
E_s = (1 - P_t)T + \sum_{i=1}^{n} P_{s,i} P_{r,i} + \sum_{r=2}^{n} \sum_{1 \leq k_r < \cdots < k_r \leq n} P_{c,k_r \cdots k_r} T_{c,k_r \cdots k_r}, \tag{8}
\]

where \( P_{s,i} = \tau_i \prod_{j \neq i} (1 - \tau_j) \) is the probability station \( i \) successfully transmits; \( T_{c,k_r \cdots k_r} \) is the time taken for a successful transmission from station \( i \); \( P_{c,k_r \cdots k_r} = \prod_{i=1}^{n} \tau_k \prod_{j \neq k_1 \cdots k_r} (1 - \tau_j) \), the probability that only the stations labeled \( k_1 \) to \( k_r \) experience a collision by attempting transmission; \( T_{c,k_r \cdots k_r} \) is the time taken for a collision from stations labeled \( k_1 \) to \( k_r \); \( P_t = 1 - \prod_{i=1}^{n} (1 - \tau_i) \) is the probability at least one station attempts transmission; and \( \sigma \) is the slot-time. For example, using the basic 802.11b MAC values found in Table I with payloads \( E_i \),

\[
\text{Header} = \text{PLCP+MAC+CRC+IP},
\]

\[
T_{c,i} = E_i + \text{Header} + \delta + \text{SIFS+ACK+PLCP+DIFS},
\]

\[
T_{c,i} = E_i + \delta + \text{Header} + \text{SIFS+ACK+DIFS},
\]

where for 802.11b ACKTimeout is the time taken for an ACK plus PLCP plus IP, \( T_{c,i} = T_{c,i} \).

The normalized throughput of the system is then \( S = \sum_{i=1}^{n} S_i \), with \( S_i = P_{s,i} E_i / E_s \) and \( E_i \) is the time spent transmitting payload data for source \( i \). Thus in order to determine the throughput and collision probability for each station, and the overall throughput, one first solves equations (7) using equations (5) and (6). Then one uses equations (8), and the expressions for \( S_i \) and \( S \).

References


