Verification of Common 802.11 MAC Model Assumptions

David Malone and Ian Dangerfield and Doug Leith

Hamilton Institute, NUI Maynooth, Ireland.*

Abstract. There has been considerable success in analytic modeling of the 802.11 MAC layer. These models are based on a number of fundamental assumptions. In this paper we attempt to verify these assumptions by taking careful measurements using an 802.11e testbed with commodity hardware. We show that the assumptions do not always hold but our measurements offer insight as to why the models may still produce good predictions. To our knowledge, this is the first in-detail attempt to compare 802.11 models and their assumptions with experimental measurements from an 802.11 testbed. The measurements collect also allow us to test if the basic MAC operation adhere to the 802.11 standards.

1 Introduction

The analysis of the 802.11 CSMA/CA contention mechanism has generated a considerable literature. Two particularly successful lines of enquiry are the use of pure p-persistent modeling (e.g. [3]) and the per-station Markov chain technique (e.g. [2]). Modeling usually involves some assumptions, and in this respect models of 802.11 are no different. Both these models assume that transmission opportunities occur at a set of discrete times. These discrete times correspond to the contention counter decrements of the stations, equivalent to state transitions in the models, and result in an effective slotting of time. Note that this slotting based on MAC state transitions is different from the time slotting used by the PHY. A second assumption of these models is that to a station observing the wireless medium, every slot is equally likely to herald the beginning of a transmission by one or more other stations. In the models this usually manifests itself as a constant transmission or collision probability.

In this paper we will show detailed measurements collected from an experimental testbed to study these assumptions. This is with a view to understanding the nature of the predictive power of these models and to inform future modeling efforts. The contribution of this paper includes the first published measurements of packet collision probabilities from an experimental testbed and their comparison with model predictions and the first detailed comparison of measured and predicted throughputs over a range of conditions.

We are not the first to consider the impact of model assumptions. In particular, the modeling of 802.11e has required the special treatment of slots immediately after a transmission in order to accommodate differentiation based

* This work was supported by Science Foundation Ireland grant IN3/03/I346.

on AIFS (e.g. $[1, 9, 11, 6, 4]$). In [13] the nonuniform nature of slots is used to motivate an 802.11e model that moves away from these assumptions.

2 Test Bed 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 [7] and one desktop PC acting as client stations. The PC acting as a client records delay measurements and retry attempts for each of its packets, but otherwise behaves as an ordinary client station. All systems are equipped with an Atheros AR5215 802.11b/g PCI card with an external antenna. All stations, including the AP, use a Linux 2.6.8.1 kernel and a version of the MADWiFi [8] wireless driver modified to allow us to adjust the 802.11e CWmin, AIFS 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 data 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 measurements, including numbers of retries and MAC-level service time. A PC is used 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.

Several software tools are used within the testbed to generate network traffic and collect performance measurements. To generate wireless network traffic we use mgen. We will often use Poisson traffic, as many of the analytic models make independent or Markov assumptions about the system being analysed. 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. Network management and control of traffic sources is carried out using ssh over the wired network.

3 Collision Probability and Packet Timing Measurement

Our testbed makes used of standard commodity hardware. In [5] we developed a measurement technique that only uses 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. For the measurements reported here, we have refined the technique described in [5] by making use of a timer in the Atheros card that timestamps the moment completed transmit descriptors are DMAed to host memory. This allows us to avoid inaccuracies caused by interrupt latency/jitter. As will be shown later, in this way we are able to take measurements with microsecond-level timing accuracy.

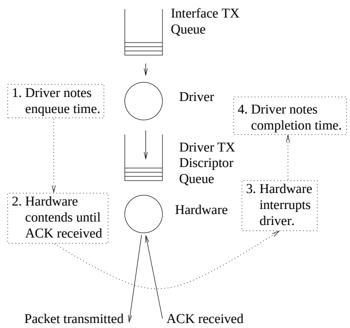


Fig. 1. Schematic of delay measurement technique.

To measure packet collision probabilities, we make use of the fact that the transmit descriptors also report the number of retry attempts R_i for each packet. Using this we can estimate the calculate the total number of retries R and the average collision probability $R/(P + R)$ where P is the number of successful packet transmissions. We can also generalist this to get the collision probability at the n^{th} transmission attempt as

$$\frac{\#\{\text{packets with } R_i \geq n\}}{\#\{\text{packets with } R_i = n\} + \#\{\text{packets with } R_i \geq n\ \}}. \tag{1}$$

This assumes that retransmissions are only due to collisions and not due to errors. We can estimate the error rate by measuring the retransmissions in a network with one station. In the environment used, the error rate is $< 0.1\%$.

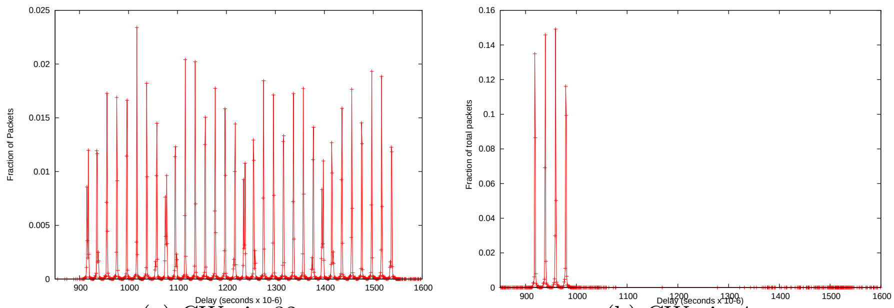
4 Validation

All the models we study assume that the 802.11 backoff procedure is being correctly followed. The recent work of [12], demonstrates that some commercial 802.11 cards can be significantly in violation of the standards. In particular, it has been shown that some cards do not use the correct range for choosing backoffs or do not seem to back off at all. We therefore first verify that the cards that we use perform basic backoffs correctly, looking at CWmin (the range of the first backoff in slots), AIFS (how many slots to pause before the backoff counter may be decremented) and TXOP (how long to transmit for).

To do this we measure the MAC access delay. This is the 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. In contrast to [12], which makes use of custom hardware to perform measurements of access delay, here we exploit the fine grained timing information available using the measurement technique described in the previous section to make access delay measurements using only standard hardware.

To test the basic backoff behaviour of the cards, we transmitted packets from a single station with high-rate arrivals and observed the MAC access delay for each packet. Figure 2(a) shows a histogram of these times to a resolution of $1\mu s$ for over 900,000 packets. We can see 32 sharp peaks each separated by the slot time of $20\mu s$, representing a CWmin of 32. This gives us confidence that the card is not subject to the more serious problems outlined in [12].

There is jitter, either in the backoff process or in our measurement technique. However, we can test the hypothesis that this is a uniform distribution by binning the data into buckets around each of the 32 peaks and applying the chi-squared test. The resulting statistic is within the 5% level of significance.

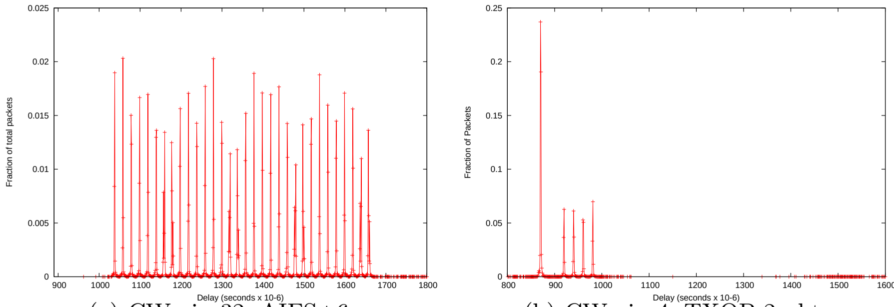


(a) CWmin 32 (b) CWmin 4

Fig. 2. Distribution of transmission times for packets with a single station. Note there a number peaks corresponding to CWmin.

The cards in question are 802.11e capable and so for comparison we adjust CWmin so that backoffs are chosen in the range 0 to 3. The results are shown in Figure 2(b) where we can see 4 clear peaks, as expected. We also see a small number of packets with longer transmission times. The number of these packets is close to the number of beacons that we expect to be transmitted during our measurements, so we believe that these are packets delayed by the transmission of a beacon frame.

Figure 3(a) shows the impact of increasing AIFS on MAC access time. In the simple situation of a single station, we expect increasing AIFS to increase MAC



(a) CWmin 32, AIFS+6 (b) CWmin 4, TXOP 2 pkts

Fig. 3. Distribution of packet transmission times for a single station. On the left, AIFS has been increased, so the peaks are shifted by $120\mu s$. On the right, TXOP has been set to allow two packets to be transmitted every time a medium access is won, so we see approximately half the packets being transmitted in a shorter time.

access times by the amount which AIFS is increased by. Comparing Figure 2(a) and Figure 3(a) confirms this.

Similarly, we can use TXOP on the cards to transmit bursts of packets, only the first of which must contend for channel access. Figure 3(b) shows the distribution of transmission times when two packet bursts are used. We see that half the packets are transmitted in a time almost $50\mu s$ shorter than the first peak shown in Figure 2(b).

These measurements indicate that a single card's timing is quite accurate and so capable of delivering transmissions timed to within slot boundaries. In this paper we do not verify if multiple cards synchronise sufficiently to fully validate the slotted time assumption.

5 Collision Probability vs Backoff Stage

Intuitively, the models that we are considering are similar to mean-field models in physics. A complex set of interactions are replaced with a single simple interaction that should approximate the system's behaviour. For example, by using a constant collision probability given by $p = 1 - (1 - \tau)^{n-1}$, where τ is the probability a station transmits, regardless of slot, backoff stage or other factors.

This assumption is particularly evident in models based on [2] as we see the same probability of collision used in the Markov chain at the end of each backoff stage. However similar assumptions are present in other models. It is the collision probability at the end of each backoff stage that we will consider in this section.

We might reasonably expect these sort of assumptions to better approximate the network when the number of stations is large. This is because the backoff stage of any one station is then a small part of the state of the network. Conversely, we expect that a network with only a small number stations may provide a challenge to the modeling assumptions.

Figure 4(a) shows measured collision probabilities for a station in a network of two stations. Each station has Poisson arrivals of packets at the same rate. We show the probability of collision on any transmission, the probability of collision at the first backoff stage (i.e. the probability of a collision on the first transmission attempt for a given packet) and the probability of collision at the second backoff stage (i.e. the probability of collision at the second transmission attempt for a given packet, providing the first attempt was unsuccessful). Error bars are conservatively estimated for each probability using $1/\sqrt{N}$, where N is the number of events used to estimate the probability.

The first thing to note is that the overall collision probability is very close to the collision probability for the first backoff stage alone. This is because collisions are overwhelmingly at the first backoff stage: to have a collision at a subsequent stage a station must have a first collision and then a second collision, but we see that less than 4% of colliding packets have a collision at the second stage.

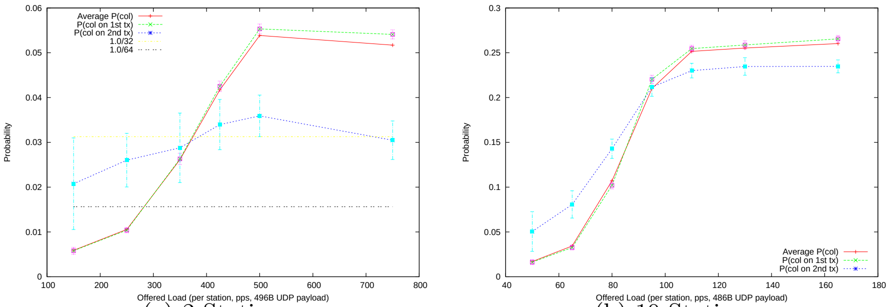
As we expect, both overall collision probability and first state collision probability increase as the offered load is increased. However, we observe that collisions at the second backoff stage show a different behaviour. Indeed, within the range of the error bars shown, this probability is nearly constant with offered load.

This difference in behaviour can be understood in terms of the close coupling of the two stations in the system. First consider the situation when the load is low. On a station's first attempt to transmit a packet, the other station is unlikely to have a packet to transmit and so the probability of collision is very low. Indeed, we would expect that the chance of collision to become almost zero as the arrival rate becomes zero.

Now consider the second backoff stage when the load is low. As we are beginning the second backoff attempt, the other station must have had a packet to transmit to have caused a collision in the first place. So, it is likely that both stations are on their second backoff stage. Two stations beginning a stage-two backoff at the same time will collide on their next transmission with probability $1/(2 * CWmin) = 1/64$ (marked on Figure 4(a)). If there is no collision, it is possible that the first station to transmit will have another packet available for transmission, and could collide on its next transmission, however as we are considering a low arrival rate, this should not be common.

On the other hand, if the load is heavy, it is highly likely that the other station has packets to send, regardless of backoff stage. This explains the increasing trend in all the collision probabilities shown. However, at the second backoff stage we know that both stations are have recently doubled their CW value. These larger than typical CW values result in smaller collision collision probability, and so we expect a lower collision rate on the second backoff stage compared to the first.

Figure 4(b) shows the same experiment, but now conducted with 10 stations in the network. Here, explicitly reasoning about the behaviour of the network is more difficult, but we see the same trends as for 2 stations: the first-stage and overall collision probabilities are very similar; collision probabilities increase as



(a) 2 Stations (b) 10 Stations

Fig. 4. Measured collision probabilities as offered load is varied. Measurements are shown of the average collision probability (the fraction of transmission attempts resulting in a collision), the first backoff stage collision probability (the fraction of first transmission attempts that result in a collision) and the second backoff stage collision probabilities (the fraction of second transmission attempts that result in a collision).

the load increases; collision probabilities at the second stage are higher than at first stage when the load is low, but vice versa when the load is high. The relative values of the collision probabilities are closer than in the case of 2 stations, but the error bars suggest they are still statistically different.

In contrast to the relatively gradual increase for two stations, we see a much sharper increase for 10 stations. Accurately capturing any sharp transition can be a challenge for a model.

In summary, while analytic models typically assume that the collision probability is the same for all backoff stages, our measurements indicate that this is generally not the case. However, collisions are dominated by collisions at the first backoff stage, and so the overall collision probability is a reasonable approximation to this. Adjustments to later-stage collision probabilities would represent second-order corrections when calculating mean-behaviour quantities (e.g. long term throughput). However, based on these measurements it is not clear if distributions or higher-order statistics, such as variances, predicted by existing models will always accurately reflect real networks.

6 Saturated Network Relationships

In this section we will consider the relationship between the average collision probability and the transmission probability. The relationship between these quantities plays a key role in many models, where it is assumed that

$$p = 1 - (1 - \tau)^{n-1}. \tag{2}$$

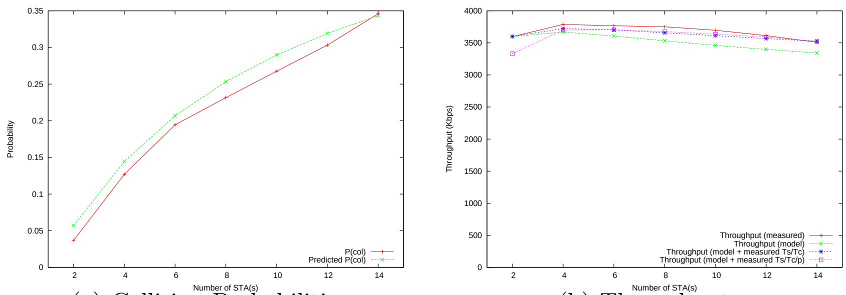
Models will typically calculate τ based on mean backoff window or use a self-consistent approach, where a second relationship between p and τ gives a pair of equations that can be solved for both.

Once τ is known, the throughput of a system is usually calculated by calculating the the average time spent transmitting payload data in a slot by the average length of a slot. That is,

$$S = \frac{E_p n\tau(1-\tau)^{n-1}}{\sigma(1-\tau)^n + T_s n\tau(1-\tau)^{n-1} + (1-(1-\tau)^n - n\tau(1-\tau)^{n-1})T_c}. \tag{3}$$

Here E_p is the time spent transmitting payload, σ is the time between counter decrements when the medium is idle, T_s is the time before a counter decrement after a successful transmission begins and T_c is the time before a counter decrement after a collision begins.

The pair of equations 2 and 3 are based on assuming that each station transmits independently in any slot. These equations can be tested independently of the rest of the model based on our measurements. Specifically, using our measurements of collision probability p, we may derive τ using equation 2 and then compare the predicted throughput given by equation 3 to the actual throughput.



(a) Collision Probabilities (b) Throughputs

Fig. 5. Predicted and measured collision probability (left) and throughput (right) in in a network of saturated stations as the number of stations is varied.

Figure 5(a) shows the predictions made by a model described in [10] for the collision probabilities in a network of saturated stations and compares them to values measured in our testbed. We see that the model overestimates the collision probabilities by a few percent.

Figure 5(b) shows the corresponding measured throughput, together with model-based predictions of throughput made in several different ways. First, p and τ are predicted using the model described in [10] and throughput derived using equation 3. We take values $T_s = 907.8\mu s$ and $T_c = 963.8\mu s$ which would be valid if the 802.11b standard was followed exactly. It can be seen that, other than for very small numbers of stations, the model prediction consistently underestimates the throughput by around 10%.

Further investigation reveals that the value used for T_c appears to significantly overestimate the T_c value used in the hardware. While the standard requires that, following a collision, stations must pause for the length of time it would take to transmit an ACK at 1Mbps our measurements indicate that the hardware seems to resume the backoff procedure more quickly. In particular, values of $T_s = 916\mu s$ and $T_c = 677\mu s$ are estimated from test bed measurements. Using once again the model values for p and τ, but now plugging in our measured values for T_s and T_c, we see in Figure 5 that this produces significantly better throughput predictions, suggesting that the estimated values for T_s and T_c are probably closer to what is in use. In particular, we note that for larger numbers of nodes, where collisions are more common, the estimated throughput now closely matches the measured throughput.

Finally, instead of predicting p using a model, we use the measured value of p and estimate τ using equation 2. We continue to use the values of T_s and T_c based on testbed measurements. We can see from Figure 5 that for larger numbers of stations the throughput predictions are very similar to the previous situation. This suggests that equation 3 is rather insensitive to the small discrepancies seen in Figure 5 for larger numbers of stations. However, for two stations we see a significantly larger discrepancy in throughput prediction. This may indicate that the independence assumptions made by equations 2 and 3 are being strained by the strongly coupled nature of a network of two saturated stations.

7 Conclusion

In this paper we have investigated a number of common assumptions used in modeling 802.11 using an experimental testbed. We present the first published measurements of conditional packet collision probabilities from an experimental testbed and compare these with model assumptions. We also present one of the first detailed comparison of measured and predicted behaviour.

We find that collision probabilities are not constant when conditioned on a station's backoff stage. However, collisions are dominated by collisions at the first backoff stage, and so the overall collision probability is a reasonable approximation to this. Adjustments to later-stage collision probabilities would represent second-order corrections when calculating mean-behaviour quantities (e.g. long term throughput). However, based on these measurements it is not clear if distributions or higher-order statistics, such as variances, predicted by these models will always accurately reflect real networks.

We also find that throughput predictions are somewhat insensitive to small errors in predictions of collision probabilities when a moderate number of stations are in a saturated network. In all our tests, we see that two station networks pose a challenge to the modeling assumptions that we consider.

In future work we may explore the level of synchronisation between stations, the effect of more realistic traffic on the assumptions we have studied and the impact of non-fixed collision probabilities on other statistics, such as delay.

References

1. R Battiti and B Li. Supporting service differentiation with enhancements of the IEEE 802.11 MAC protocol: models and analysis. Technical Report DIT-03-024, University of Trento, 2003.
2. G Bianchi. Performance analysis of IEEE 802.11 Distributed Coordination Function. *IEEE JSAC*, 18(3):535–547, 2000.
3. F Cali, M Conti, and E Gregori. IEEE 802.11 wireless LAN: Capacity analysis and protocol enhancement. In *Proceedings of IEEE INFOCOM, San Francisco, USA*, pages 142–149, 1998.
4. P Clifford, K Duffy, J Foy, DJ Leith, and D Malone. Modeling 802.11e for data traffic parameter design. In *WiOpt*, 2006.
5. I Dangerfield, D Malone, and DJ Leith. Experimental evaluation of 802.11e edca for enhanced voice over wlan performance. In *International Workshop On Wireless Network Measurement (WiNMee)*, 2006.
6. P Engelstad and ON Østerbø. Queueing delay analysis of IEEE 802.11e EDCA. In *IFIP WONS*, 2006.
7. Soekris Engineering. `http://www.soekris.com/`.
8. Multiband Atheros Driver for WiFi (MADWiFi). `http://sourceforge.net/projects/madwifi/`. r1645 version.
9. Z Kong, DHK Tsang, B Bensaou, and D Gao. Performance analysis of IEEE 802.11e contention-based channel access. *IEEE JSAC*, 22(10):2095–2106, 2004.
10. D Malone, K Duffy, and DJ Leith. Modeling the 802.11 Distributed Coordination Function in non-saturated heterogeneous conditions. *To appear in IEEE ACM T NETWORK*, 2007.
11. JW Robinson and TS Randhawa. Saturation throughput analysis of IEEE 802.11e Enhanced Distributed Coordination Function. *IEEE JSAC*, 22(5):917–928, 2004.
12. A Di Stefano, G Terrazzino, L Scalia, I Tinnirello, G Bianchi, and C Giaconia. An experimental testbed and methodology for characterizing IEEE 802.11 network cards. In *International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2006.
13. Ilenia Tinnirello and Giuseppe Bianchi. On the accuracy of some common modeling assumptions for EDCA analysis. In *International Conference on Cybernetics and Information Technologies, Systems and Applications*, 2005.