A Hardware Test Bed for Measuring IEEE 802.11g Distribution Coordination Function Performance

Andrew Symington
Computing Laboratory
University of Oxford
Oxford, United Kingdom
Email: andrew.symington@comlab.ox.ac.uk

Pieter Kritzinger
Department of Computer Science
University of Cape Town
Cape Town, South Africa
Email: psk@cs.uct.ac.za

***Abstract*—The Distributed Coordination Function is one of three channel access control protocols specified by the IEEE 802.11 standard. In this paper we present a method of measuring DCF performance using a test bed built with off-the-shelf hardware. Performance is measured by normalized aggregate throughput as a function of the number of stations contending for channel access. We present measurements for both basic access and RTS/CTS access in fully-connected IEEE 802.11g networks experiencing conditions of saturation. We compare our measurements to results from three analytic models and a simulator, all of which shared the same assumptions about the workload model and operation of DCF. For small networks the analytic models predict a much lower performance than shown through simulation and test bed experiments. As the network grows, so the measured performance deteriorates significantly faster than predicted by the analytic models. We attribute this to inaccuracies in the analytic model, imperfect channels and queuing. The simulation results fit the measured data with more accuracy, as the simulator makes fewer restrictive assumptions about DCF when compared to the analytic models. This is the first paper to provide a cross-comparison of test bed, simulation and analytic results for IEEE 802.11g DCF performance.**

I. Introduction

The prolific adoption of wireless local area network (WLAN) technology over the last decade has been driven by decreasing hardware and maintenance costs, as well as improved radio technology. The mobility and deployment advantages of WLANs are generally offset by bandwidth restrictions; radio channels are shared amongst many users and therefore access must be carefully coordinated. Every WLAN standard must specify a channel access control mechanism, which defines how stations coordinate their transmissions on the shared wireless channel. In IEEE 802.11 two distributed channel access control protocols are defined, the Distributed Coordination Function (DCF) and Enhanced Distributed Channel Access (EDCA).

Although the main focus of this paper is on DCF performance, it is important at this point to observe the close relationship between EDCA and DCF. In essence, EDCA is a compatible successor to DCF that primarily provides support for traffic class differentiation through assymetric queuing, but also adds several performance enhancements to the protocol. Both EDCA and DCF share the same binary exponential back-off mechanism that will be outlined in Section II-A.

In this paper we measure the performance of DCF using a nine station IEEE 802.11g test bed built from off-the-shelf hardware. Here, performance is measured solely by normalized aggregate throughput, which is the proportion of channel time attributed to transmitting useful data bits. We assume full-connectivity between stations and a saturation workload model with fixed packet length. We compare the test bed measurements to widely-accepted analytic and simulation results, both of which make the same model assumptions. We show that analytic models are fairly pessimistic for small networks and too optimistic for larger networks.

Section II presents the operation of the DCF protocol, focusing specifically on the binary back-off mechanism and framing. Section III discusses existing approaches for measuring IEEE 802.11 DCF performance. In Section IV we report on the methods and materials used to construct the test bed. We also discuss the experiments that we conducted and list the important test bed parameters that we used. In Section V we present test bed measurements and contrast them with analytic predictions and simulation results. Finally, Section VI concludes this paper and proposes future research directions.

II. Distributed Coordination Function

A. Binary exponential back-off

DCF is a best-effort contention-based protocol which uses exponential binary back-off to coordinate access to a wireless channel. Time is discretised into slots of fixed length σ microseconds and, at the start of every slot, each station performs a *Clear Channel Assessment* (CCA) to determine whether or not the channel is clear. When new data arrives for transmission the station selects a random number uniformly in the discrete interval $[0, CW_{min} - 1]$, where CW_{min} is a system parameter. The DCF protocol dictates that the station must wait, or *back-off*, for the chosen number of slots. If a CCA reports busy at any time during this back-off counter halts until the medium is perceived as idle for a DCF Inter-Frame Space (DIFS). The length of a DIFS is chosen such that acknowledgment (ACKs) frames take priority over new MAC Protocol Data Unit (DATA) frames. Values for CW_{min}, CW_{max}, Short Inter-Frame Space (SIFS) and σ are determined by the IEEE 802.11 physical layer (PHY) and listed for common PHYs in Table I. The DIFS period is always equal to

TABLE I
PARAMETERS FOR IEEE 802.11

Parameter	802.11b	802.11a	*Mixed* 802.11g	*Pure* 802.11g
Slot time	$20\mu s$	$9\mu s$	$20\mu s$	$9\mu s$
SIFS	$10\mu s$	$16\mu s$	$10\mu s$	$10\mu s$
DIFS	$50\mu s$	$34\mu s$	$50\mu s$	$28\mu s$
CW_{min}	16	32	32	32
CW_{max}	1024	1024	1024	1024

$2\sigma + SIFS$. Once the DATA frame has been transmitted the sender awaits an ACK from the receiver, which is expected back within a $SIFS + \delta$ period, where δ is the microsecond propagation delay. On failure, the station recalculates the contention window, backs-off for a new number of slots and finally attempts retransmission. On the i^{th} failure the new contention window is $[0, CW_i - 1]$ with CW_i given by Equation 1. The maximum retry count for basic access is 5, after which the frame is dropped.

$$CW_i = min(2^i CW_{min}, CW_{max}) \tag{1}$$

Transmission failures in basic access are costly, since they can only be detected after a full DATA frame is forwarded. Under a high load or in multi-hop networks with many hidden nodes this problem can reduce aggregate performance substantially. An optional RTS/CTS access mechanism is therefore also specified in IEEE 802.11, in which the sender issues a Request-to-Send (RTS) frame prior to transmitting the DATA frame. The intended receiver responds with a Clear-to-Send (CTS) frame within a $SIFS + \delta$ period, after which transmission continues as per basic access. The purpose of the RTS and CTS frames is to inform neighbouring stations of an upcoming transmission. The neighbours then refrain from accessing the channel for the transmission time, which is specified in the duration field of both frames. RTS/CTS access reduces the cost of collision by adding a small amount of overhead. Therefore, in practice the access mechanism depends on the length of the data frame and the RTS threshold. The maximum retry count for RTS/CTS access is 7.

We denote the time required to send a frame of type X as $T[X]$. Regardless of the access mechanism used, on observing a collision all stations defer access to the channel for an Extended Inter-Frame Space (EIFS) period, which is equal to $T[ACK] + SIFS + \delta$. In multi-hop networks collisions do not necessarily occur uniformly across the network, so this EIFS period provides sufficient time for a hidden receiver to acknowledge the DATA frame. The total time required to transmit with success T_s or collision T_c, for both basic and RTS/CTS access, is given by the following four equations.

$$\begin{aligned}
T_s^{bas} &= T[DATA] + SIFS + T[ACK] + DIFS + 2\delta \\
T_s^{rts} &= T[RTS] + T[CTS] + 2(SIFS + \delta) + T_s^{bas} \\
T_c^{bas} &= T[DATA] + DIFS + \delta \\
T_c^{rts} &= T[RTS] + DIFS + \delta
\end{aligned}$$

Note that these equations do not account for the post-collision EIFS period, as they are derived from the perspective of a transmitting station.

B. Frames in pure IEEE 802.11g networks

IEEE 802.11g unicast DATA frames are transmitted at the *data rate* (a rate supported by both the sender and receiver) while multicast control frames (RTS,CTS and ACK) are transmitted at the *basic rate* (a rate supported by all stations). Both the basic and data rate are chosen according to station capabilities in conjunction with some *rate control algorithm* that dynamically adjusts the rate according to channel conditions. In this paper we consider pure IEEE 802.11g networks in which only Orthogonal Frequency Division Multiplexing (OFDM) rates are supported[1]. This dictates that the maximum data rate is 54Mbps, while the maximum basic rate is 24Mbps.

For any frame that is sent, the PHY begins by transmitting a preamble to synchronize the transmitter and receiver. This is followed immediately by a 4 microsecond signal header, which is made up of 24 bits sent at 6Mbps. The signal header contains the length of the upcoming payload, as well as the rate at which it will be sent. Before transmitting the data the PHY appends a 16 bit service field and 6 bit tail field to the MAC Protocol Data Unit (MPDU). A variable number of padding bits are also added to the payload to ensure that it is a perfect multiple of the block size required by the coding rate. The PHY payload is then transmitted at either the data or basic rate, depending on the frame type. Finally, the payload is followed by a 6 microsecond signal extension to make OFDM timings similar to IEEE 802.11a. Figure 1 shows the time required to forward RTS, CTS and ACK frames, as well as a DATA frame of 1000 bytes for such a network.

III. RELATED WORK

Bianchi's [1] widely-accepted analytic model for DCF calculates aggregate normalized throughput for a network of n fully-connected stations under saturation conditions. The back-off process is modelled as a discrete-time Markov chain with time unit equal to the slot time. Using this model in conjunction with assumption that all stations contend equally for channel access, the probability for channel transmission in any arbitrary slot P_t and the probability of successful transmission P_s are derived. Normalized aggregate throughput S is calculated as the proportion of channel time attributed to transmitting data bits. Equation 2 shows this relationship, with $T^*[DATA]$ being the time taken to transmit only data bits.

$$S = \frac{P_s P_t T^*[DATA]}{(1 - P_t)\sigma + P_t\left(P_s T_s + (1 - P_s)T_c\right)} \tag{2}$$

Bianchi's model assumes that the probability of collision is constant and that the back-off counter does not suspend while the medium is in use. Although the model also does not account for the post-collision EIFS period, its inclusion

[1]Mixed mode networks contain both IEEE 802.11b and IEEE 802.11g stations. In such a scenario the slot time is 20 microseconds and the basic rate is chosen as a non-OFDM rate supported by all devices

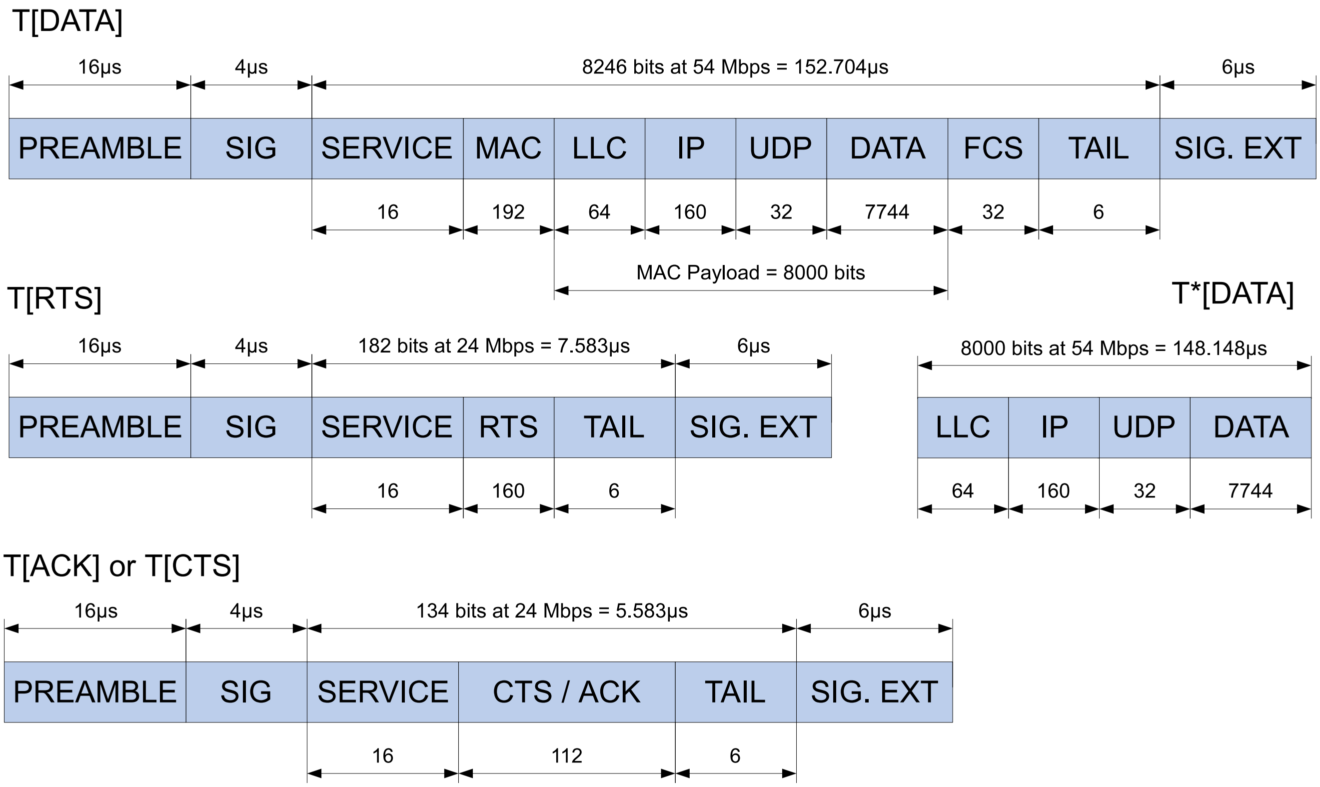


Fig. 1. Transmission time for an IBSS DATA frame (top), RTS frames (middle) and CTS or ACK frames (bottom)

involves a simple change to one of the model's variables. The authors Ziouva and Antonakopoulos [2] as well as Vishnevsky and Lyakhov [3] present independent extensions to Bianchi's model, both of which attempt to capture the effect of back-off counter seizing on aggregate throughput. The former model assumes that the number of time units seized is geometrically distributed.

Bianchi's model assumes further that there are an infinite number of retries on the final stage of back-off. Wu *et al* [4] present an extension to Bianchi's model that implements the retry mechanism used in the standard. In a subsequent paper [5] Bianchi acknowledged several extensions to the original model and presented an approach to model delay using Little's theorem.

Other authors have proposed extensions to the original model for EDCA [6], [7], noisy channels [8], [9], [10], non-saturated Poisson traffic [11] or variable packet length [12]. Chatzimisios proposes a delay analysis for DCF, first in [13] for Bianchi's original model and, later, for finite retries in [14]. Szczypiorski and Lubacz [15] present a unified analytic model for DCF in IEEE 802.11g networks with back-off seizure, noisy channels and finite retransmissions

Several different simulators for DCF exist. Weinmiller *et al.* [16] developed a process-oriented simulation model. Chen *et al.* [17] implemented an activity scanning model for NS-2. Cocorada [18] used the OMNeT++ simulation environment to model the IEEE 802.11g standard using the discrete-event simulation paradigm. Bianchi and Tinnirello [5], as well as Kritzinger *et al* [19] implement independent event-driven simulators for EDCA and DCF respectively. In this paper we consider a test bed with the same physical parameters (SIFS, DIFS, slot time) as the simulation with the exception that it makes no assumptions about the behaviour of the wireless channel.

Several academic IEEE 802.11 test beds exist, such as the Orbit Laboratory [20] and the MIT Roofnet [21] project. However, no conclusive research has been conducted to measure DCF performance and compare the results to equivalent analytic models and simulation results.

IV. Methods and Materials

A. Hardware test bed

1) Design: Our test bed, depicted in Figure 2, comprises of nine disk-less *stations* and one *central controller*, all of which are connected to a wired Ethernet control backbone. The control backbone is used to manage experiments and perform maintenance tasks. Each station has a single Atheros AR5212[2] IEEE 802.11g card, over which experiments were conducted to measure the performance of DCF.

On booting, the client stations request a Linux kernel and root file system from the controller station over the wired

[2]The Atheros AR5212 IEEE 802.11g chipset is frequently used for research projects, because good community support is available, and (ii) complete drivers are readily available for most operating systems.

control backbone using the *Pre-boot Execution Environment* (PXE). The root file system is an embedded version of Gentoo Linux designed specifically for the test bed and associated experiments. Once the kernel boots into the root file system, a Network File System (NFS) store provides each station with a unique set of parameters, such as individual *Secure Shell* (SSH) keys.

The wireless cards are controlled by *MadWifi* drivers for Linux. Each wireless station has a detachable 2dBi antenna. We made provision for an optional attenuator between the card and antenna in order to mimic a small-scale multi-hop environment. We also raised the antenna by 600mm using plastic tubing, in order to prevent the metal chassis from interfering with radio propagation.

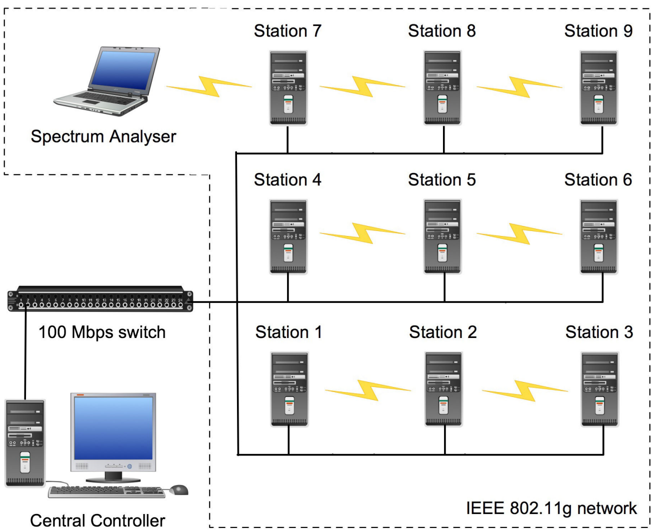


Fig. 2. The nine station hardware test bed. Station 5 is configured as the sink, while all remaining stations are potential sources. The wired backbone network is used to configure all stations and collect experiment results.

2) Driver software: On booting, the sink starts a MESHNET-LISTENER service, while the remaining eight stations start a MESHNET-CLIENT service. The purpose of MESHNET-LISTENER is to accept any incoming UDP socket requests and packets generated by any MESHNET-CLIENT instance over the wireless network. Note that MESHNET-LISTENER does not acknowledge any incoming packets as this would affect the performance tests.

The MESHNET-CONTROLLER application is run on the control station and a single configuration file specifies the high-level machine and workload model parameters for the experiments. These parameters are translated to low level configuration instructions. One of the major challenges was translating IEEE 802.11g DCF machine model instructions to MadWifi driver configuration commands (see Table II).

For each run of an experiment MESHNET-CONTROLLER selects a random subset of client stations equal to the size of the network that is being tested. It then forwards configuration instructions over the back-bone network to each station in the set, which configures the MadWifi interface correctly. Once all stations are configured the MESHNET-CONTROLLER instructs all clients to connect to the sink for a fixed duration and saturate the connection with fixed length packets. We use a interface state snapshot, provided by the MadWifi *athstats* application, immediately before and immediately after the experiment to determine how many packets were forwarded successfully by each station. Each MESHNET-CLIENT reports this value to the MESHNET-CONTROLLER application, which merges the data.

TABLE II
MADWIFI PARAMETERS AND INTERPRETATION

CONFIGURATION STRING	DESCRIPTION
iwpriv ath0 mode 11g	Set the device to IEEE 802.11g mode
iwpriv ath0 pureg 1	Only allow IEEE 802.11g rates
iwpriv ath0 protmode 0	Disable protection for IEEE 802.11b
sysctl -w dev.wifi0.slottime=9	Reduce the slot time to $9\mu s$
sysctl -w dev.wifi0.diversity=0	Turn off antenna diversity
iwpriv ath0 wmm 0	Use DCF and not EDCA
iwpriv ath0 abolt 0	Turn off proprietary protocol extensions
iwconfig ath0 rate 54M fixed	Set the data rate to 54 Mbps
iwpriv ath0 mcast_rate 24000	Set the basic rate to 24 Mbps
iwconfig ath0 mode ad-hoc	Use ad-hoc style interface
iwconfig ath0 txpower 5	Adjust the transmit power to 5 dBm
iwconfig ath0 channel 1	Use ISM channel one (2.412 GHz)

3) Mitigating interference: Many consumer devices, such as microwave ovens and Bluetooth, use the same 2.4 GHz *Industrial Scientific and Medical* (ISM) band as IEEE 802.11. Such devices may cause (i) interference and therefore frame corruption, which results in packet loss, or (ii) the IEEE 802.11 MAC to sense the channel as busy and defer access to a later stage, which causes packet delay. Interference can therefore skew experiment results, especially since the simulations and analytic models make the assumption of a perfect wireless channel. In an attempt to mitigate the effect of interference we set up the test bed at a remote location and conducted a channel noise assessment prior to experimentation.

For the first test we used a spectrum analyser to measure the energy in the full 2.4GHz ISM band. We noticed a small amount of interference from a non-802.11 device on channel six (2.437 GHz). We therefore configured the test bed to use a non-overlapping channel, centered at 2.412 GHz (channel 1). Figure 3 indicates a noise floor of around -95 dBm and a peak energy of less than -79 dBm throughout the test period, which is below the *Received Signal Strength Indication* (RSSI) threshold for the 24 Mbps OFDM rate. In order to ensure that low-power IEEE 802.11 control frames were not missed by the spectrum analyser we put the sink's wireless card into RF monitor mode. We observed no frames for the same period as the channel energy test.

4) Experiments: The objective of our experiments was to measure the change in normalized aggregate throughput as a function of the number of contending stations, all of which attempt to saturate the network with fixed size packets. We conducted two sets of experiments, one for basic access and the other for RTS/CTS access. Each set was composed of 8 different experiments, each of which measured performance

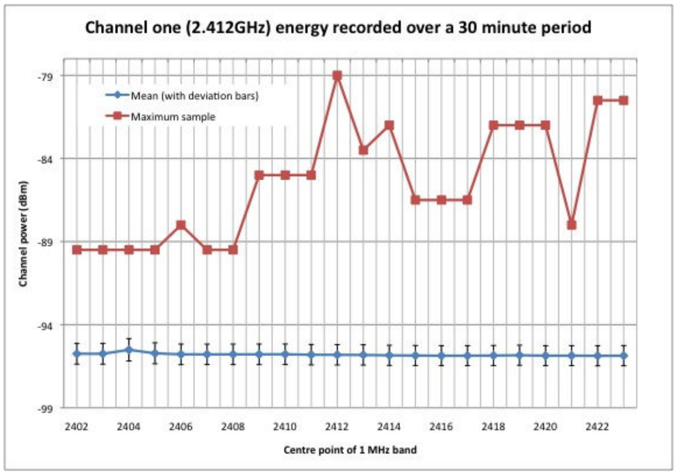


Fig. 3. The energy spectral density for Channel 1 over a 30 minute period

in a network comprising of 1 to 8 contending stations. Each experiment was repeated 30 times to calculate a mean and 95% confidence level under the assumption that results were identical and independent with a normal distribution. For every run we chose a different random subset of the client stations equal to the network size that was being tested at the time. A photograph of the test bed is shown in Figure 4.

B. Event-driven simulator

We used Bianchi and Tinnirello's [5] IEEE 802.11g event-driven EDCA simulator to derive the simulation results. To mimic DCF we used a single class of traffic with an AIFS value of zero and persistent factor[3] of two. The default MAC header was reduced to 242 bits because we consider only IBSS data frames for our experiment. We also reduced the SIFS period to 10 microseconds to be in line with the IEEE 802.11g standard. The remaining parameters were left at their default values. We fixed the data and basic rates at 54Mbps and 24Mbps respectively, and then ran the simulation for $n = 1$ to 8 contending stations. In the steady state it is not possible for any station in a network of size n to achieve a throughput greater than $\mu = \frac{54}{n} Mbps$. We therefore set the arrival rate of data at each station equal to μ, which effectively saturated the network. Separate simulation experiments were conducted for basic access and RTS/CTS access. Each network size was simulated 10 times with different starting seeds and every run lasted 10 seconds.

C. Analytic models

The analytic software from Bianchi and Tinnirello [5] makes use of a numerical method to solve for the non-linear relationship between collision probability p and per-slot transmission probability τ. The value for τ is easily calculated once p is known. Results were obtained for (i) Bianchi's [1] original model, (ii) Ziouva's [2] extension for back-off suspension and (iii) Xiao's [6] extension for both back-off suspension and finite retry counters. Parameters for the analytic models were as those outlined in Section II with $\delta = 1$ microsecond.

[3]The persistent factor affects the scaling of the contention window

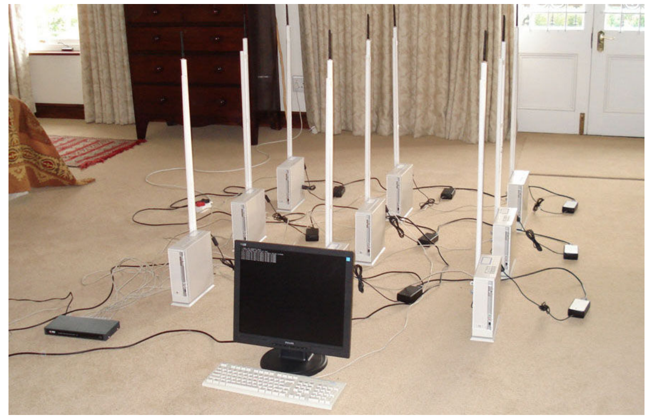


Fig. 4. The experimental set up

V. Discussion of Results

The two graphs in Figure 5 show how DCF performance measured on the test bed changes as a function of the number of stations, when compared to results from simulation and analytic models using, where possible, the same system parameter settings. We include 95% confidence intervals for both the test bed measurements and simulation results. Our comparison reveals a significant difference in results amongst the analytic models, simulation and measurements.

The three analytic models show similar trends, however Ziouva's model initially yields a relatively lower performance for basic access. For Bianchi's model the back-off counter can expire during a transmission and the contention window and therefore expands rapidly to its maximum when probability of medium availability is small. Conversely, in Ziouva's model the contention window is kept small by preventing transmission attempts during busy periods. Initially, this deferral has a negative effect on performance, but for more than six stations the probability of medium access is so low that back-off suspension becomes advantageous. In Xiao's model this advantage appears to be offset by finite retries. The main reason for the similarity amongst the analytic results is that they share the same foundational modelling assumptions. In fact, the small differences between the three models occur as a result of relatively trivial corrections to the back-off process and associated parameters.

What is more interesting is the observed difference between analytic and measured results. Again, this is related to the fact that all three analytic models share a common set of foundational assumptions. One such assumption, known as *station decoupling*, allows a single station to be decoupled from the network and its performance modelled independently. Once a solution has been found for this single station, the performance of the entire network may be more readily found. This provides an analytically tractable approach to solving for network performance; modelling the joint evolution of all stations simultaneously would result in an exploding state-space. Recently, Duffy and Malone [22] explored the decoupling assumption and concluded that the decoupling assumption holds in small saturated networks, which explained the accuracy of Bianchi's results.

For basic access the simulator calculates the performance trend with greater precision than analytic models. However, although starting higher, the measured performance declines much faster than suggested through simulation. We suspect that this is related to the fact that the simulator makes several assumptions about the wireless medium and queuing time between the PHY and MAC layers. For RTS/CTS access, however, a similar relationship is not observed.

Simulation results agree with analytic models in suggesting that aggregate network performance converges asymptotically to some upper bound for RTS/CTS access. However, measurements disagree and show that that performance declines at a decreasing rate. We suspect that the difference may be related to the manner in which simulation and the test bed handle ACK and CTS timeouts. However, notice that the measured decline is not as rapid as that of basic access, evidence of the fact that the RTS/CTS mechanism lowers collision cost.

VI. Conclusion

The work reported in this paper relates to fully-connected IEEE 802.11g DCF networks, subject to saturation conditions with fixed packet length. The novelty of the work is that we used the same system parameter values in both the test bed and the simulator and reflected the analytic model assumptions (for instance, saturation traffic) where possible. This, we believe, made it possible to directly compare the results from the three different modelling paradigms.

Our results indicate that analytic models for both basic and RTS/CTS access are pessimistic for small networks. As the network grows in size the measured performance drops more rapidly than predicted by the analytic models. We show a crossing point at which the analytic models become increasingly optimistic about DCF performance. For basic access our results indicate that simulation provides a superior fit to measured results. However, the measured performance once again drops more rapidly as subsequent stations are added to the network. We attribute this trend to queuing delays and an imperfect wireless channel.

Most importantly, however, we have shown a difference between predicted and measured performance in saturated DCF networks. Our work highlights the need for new modelling techniques - perhaps with fewer restrictive assumptions about station decoupling and the back-off process - that will aid network designers to accurately forecast system performance prior to deployment of an IEEE 802.11g DCF network.

Acknowledgment

We would like to firstly thank Professors Guiseppe Bianchi and Ilenia Tinnirello for providing access to their analytic solver software and simulator for IEEE 802.11g DCF. We'd also like to thank Dr Niki Trigoni and members of the Sensor Network Group at Oxford for supporting this paper.

References

[1] G. Bianchi, "IEEE 802.11 saturation throughput analysis," *IEEE Communications Letters*, vol. 2, no. 12, pp. 318–320, 1998.

[2] E. Ziouva and T. Antonakopoulos, "CSMA/CA performance under high traffic conditions: throughput and delay analysis," *Computer Communications*, vol. 25, pp. 313–321, 2002.

[3] V. Vishnevsky and A. Lyakhov, "IEEE 802.11 wireless LAN: Saturation throughput analysis with seizing effect consideration," *Cluster Computing*, vol. 5, pp. 133–144, 2002.

[4] H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma, "Performance of reliable transport protocol over IEEE 802.11 wireless LAN: analysis and enhancement," *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 2, pp. 599–607, 2002.

[5] G. Bianchi and I. Tinnirello, "Remarks on IEEE 802.11 DCF performance analysis," *IEEE Comms. Letters*, vol. 9, no. 8, pp. 765–767, 2005.

[6] Y. Xiao, "Performance analysis of IEEE 802.11e EDCF under saturation condition," *Communications, IEEE International Conference on*, vol. 1, pp. 170–174, 2004.

[7] J. Robinson and T. Randhawa, "Saturation throughput analysis of IEEE 802.11e enhanced distributed coordination function," *Selected Areas in Communications, IEEE Journal on*, vol. 22, no. 5, pp. 917–928, 2004.

[8] Q. Ni, T. Li, T. Turletti, and Y. Xiao, "Saturation throughput analysis of error-prone 802.11 wireless networks," *RRM for Next-Gen. Wireless and Mobile Comms. Systems*, vol. 5, no. 8, pp. 945–956, 2005.

[9] F. Daneshgaran, M. Laddomada, F. Mesiti, and M. Mondin, "Unsaturated throughput analysisof IEEE 802.11 in presence of non ideal transmission channel and capture effects," *IEEE Trans. on Wireless Comms.*, 2008.

[10] Y. Zheng and K. Lu, "Performance analysis of IEEE 802.11 DCF in imperfect channels," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 5, pp. 1648–1656, 2006.

[11] K. Duffy, D. Malone, and D. J. Leith, "Modeling the 802.11 distributed coordination function in non-saturated conditions," *IEEE Communications Letters*, vol. 9, no. 8, pp. 715–717, 2005.

[12] H. Chen and Y. Li, "Performance model of IEEE 802.11 DCF with variable packet length," *IEEE Communications Letters*, vol. 8, no. 3, pp. 186–188, 2004.

[13] P. Chatzimisios, V. Vitsas, and A. Boucouvalas, "Throughput and delay analysis of IEEE 802.11 protocol," in *Networked Appliances, IEEE 5th International Workshop on*, 2002.

[14] P. Chatzimisios, A. Boucouvalas, and V. Vitsas, "IEEE 802.11 packet delay-a finite retry limit analysis," vol. 2, Dec. 2003, pp. 950–954.

[15] K. Szczypiorski and J. Lubacz, "Saturation throughput analysis of IEEE 802.11g (erp-ordm) networks," *Personal Wireless Communications, IFIP Journal on*, vol. 245, pp. 196–205, 2007.

[16] J. Weinmiller, M. Schläger, A. Festag, and A. Wolisz, "Performance study of access control in WLANs—IEEE 802.11 DFWMAC and ETSI RES 10 Hiperlan," *Mob. Netw. Appl.*, vol. 2, no. 1, pp. 55–67, 1997.

[17] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein, "Overhaul of IEEE 802.11 modeling and simulation in NS-2," in *MSWiM '07: Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*. New York, NY, USA: ACM, 2007, pp. 159–168.

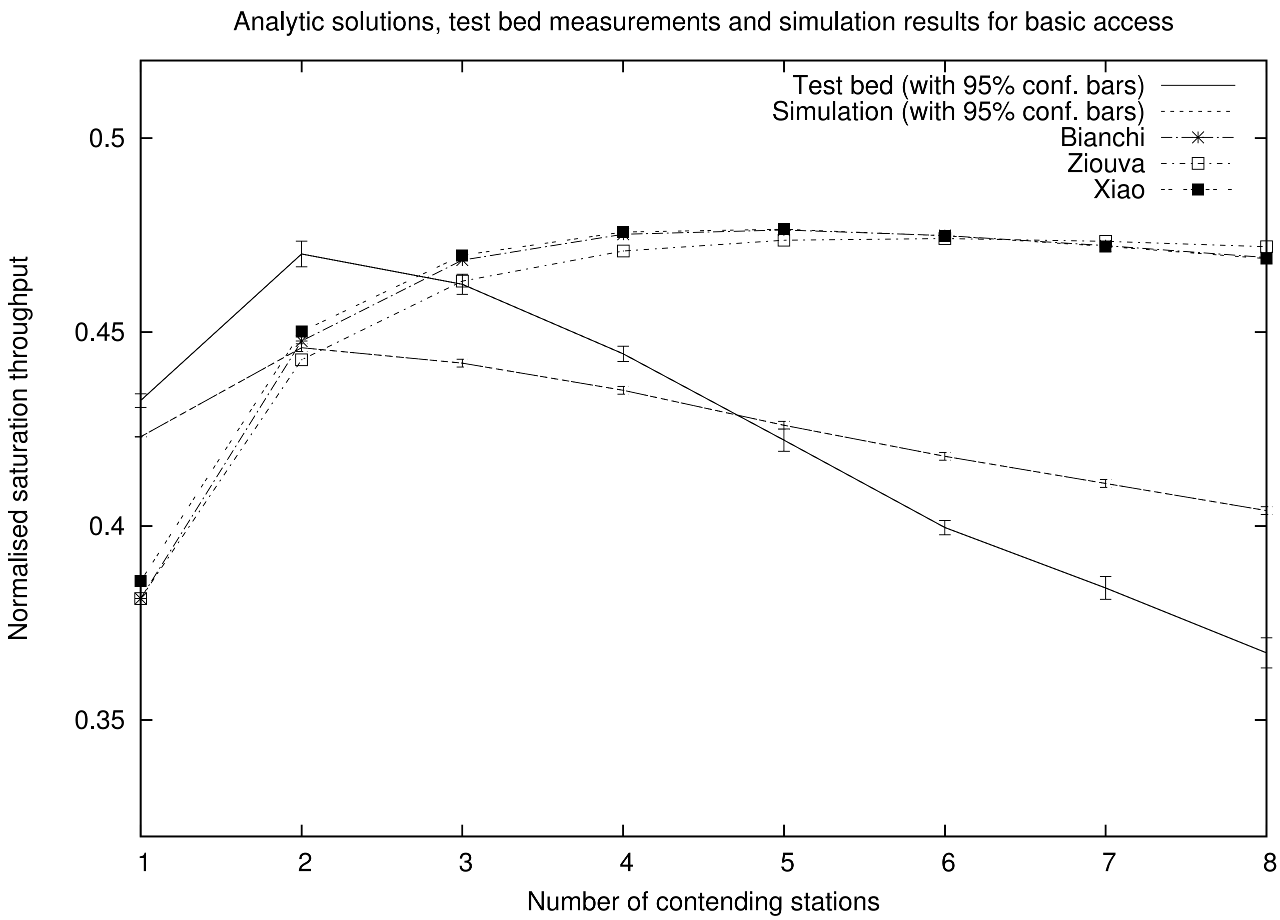
[18] S. Cocorada, "An IEEE 802.11g simulation model with extended debug capabilities," in *Simutools '08: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*. ICST, 2008, pp. 1–3.

[19] P. Kritzinger, H. Msiska, T. Mundangepfupfu, P. Pileggi, and A. Symington, "Verifying the accuracy of some IEEE802.11g DCF analytic models," Department of Computer Science, University of Cape Town, Tech. Rep. CS09-03-00, March 2009.

[20] M. Ott, I. Seskar, R. Siracusa, and M. Singh, "Orbit testbed software architecture: Supporting experiments as a service," in *IEEE Tridentcom, Trento, Italy*, 2005.

[21] D. Aguayo, J. Bicket, S. Biswas, and D. D. Couto, "MIT roofnet: Construction of a production quality ad-hoc network," in *MobiCom Poster*, 2003.

[22] K. Huang, K. Duffy, D. Malone, and D. Leith, "Investigating the validity of ieee 802.11 mac modeling hypotheses," in *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, Sept. 2008, pp. 1–6.



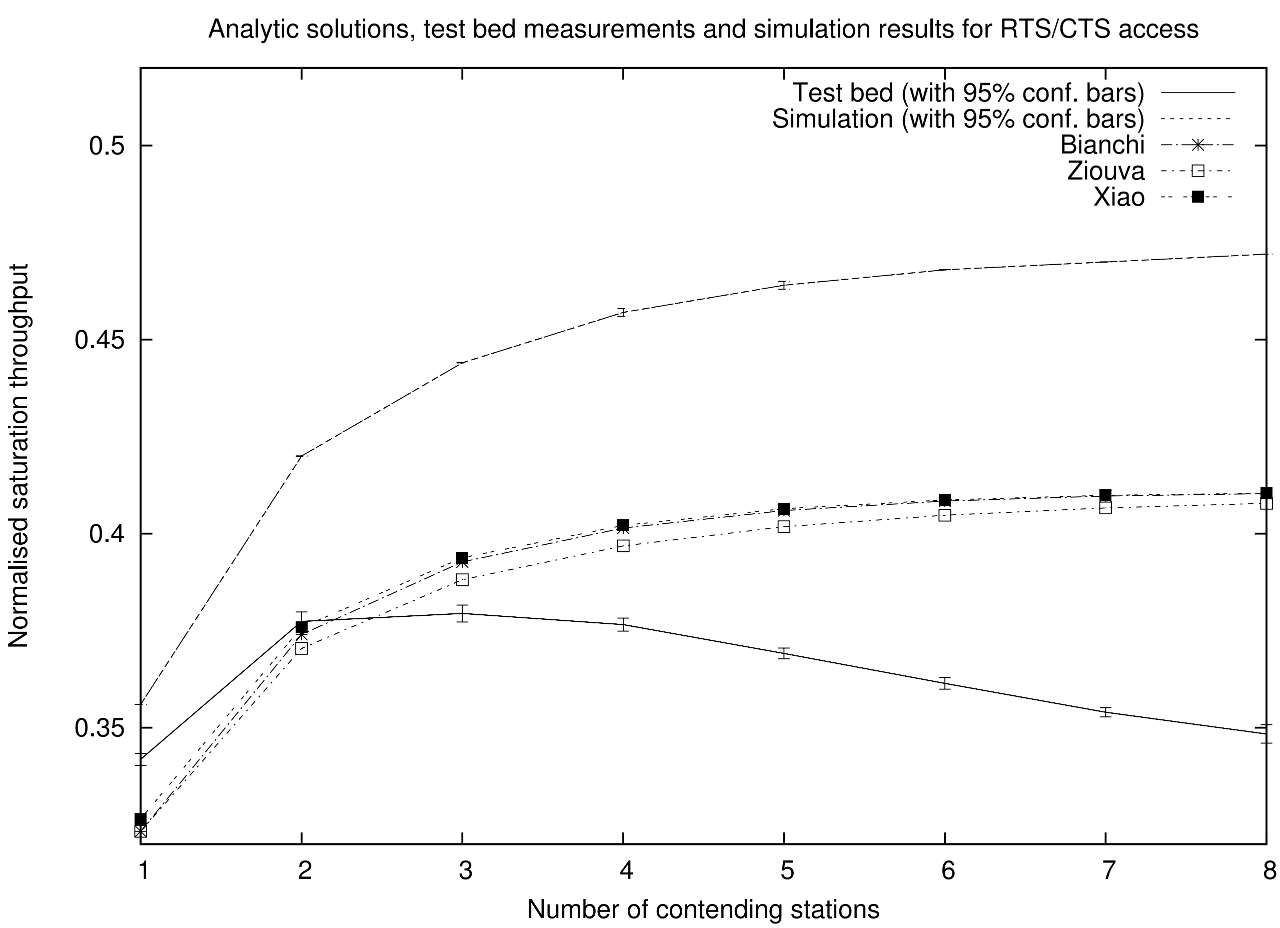


Fig. 5. Experiment results for basic access (top) and RTS/CTS access (bottom). In both experiments basic and control frames were transmitted at 54Mbps and 24Mbps respectively. The number of stations contending for channel access is shown on the x-axis and the resultant normalised throughput for the entire network is shown on the y-axis. Each simulation was run ten times and each test bed experiment was run thirty times to obtain a 95% confidence intervals for each measurement, which are shown as bands extending from the sample points.