Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computer Communications 29 (2006) 842–854

computer communications

www.elsevier.com/locate/comcom

End-to-end delay analysis and admission control in 802.11 DCF WLANs ☆

Albert Banchs *, Pablo Serrano, Arturo Azcorra

Universidad Carlos III de Madrid, Av. Universidad 30, 28911 Leganés, Madrid, Spain

Available online 12 September 2005

Abstract

In this paper, we first present an analytical model to study the distribution of the backoff delay in an 802.11 DCF WLAN under saturation conditions. We show that, with our method, the probability that the backoff delay is below a given threshold can be computed accurately and efficiently. Then, we extend our backoff delay distribution model to analyze the end-to-end delay, and propose an admission control algorithm based on this analysis. The proposed algorithm is evaluated in a mixed environment with voice and data stations, and simulation results confirm that it effectively provides voice stations with end-to-end delay guarantees. The algorithm is also evaluated in terms of computational cost and shown to be efficient enough for run-time usage.

Keywords: Wireless LAN; 802.11 DCF; Backoff delay; Delay guarantees

1. Introduction

As 802.11 WLANs see their capacity increased (from the traditional 2 Mbps channel capacity to 11 Mbps in 802.11b and 54 Mbps in 802.11a), these networks become better suited for the transport of real-time traffic. Since, the performance of real-time applications is largely dependent on delay, there arises the need for an analysis of the delay in this type of networks.

To this date, the analysis of the delay in 802.11 WLAN has received some attention [1–3]. The studies in those papers, however, are limited to the average delay, which is insufficient to assess the performance of real-time applications, as these require not only a low average delay but a low delay for all (or most of) their packets. The analyses have [4,5] overcome this limitation by introducing probability generating functions (pgf's), which allow the computation of the probability distribution function (pdf) of the delay. Yet, these methods are computationally costly and hence of little practical use, e.g. to perform admission control functionality. Furthermore, most of the previous studies are restricted to the backoff component of the delay, which is only one portion of the total delay in a WLAN. We conclude that the results from these previous works cannot be applied to provide real-time applications with their required delay guarantees via admission control.

The contribution of this paper is twofold. In the first part of the paper, an analysis of the distribution of the backoff delay under saturation conditions (hereafter referred as saturation backoff delay) is presented. By backoff delay we understand the time elapsed since a packet starts its backoff process until it is successfully transmitted.[1] With saturation conditions we mean that all the stations in the WLAN always have packets to transmit.

The second part of the paper presents, based on the *saturation backoff delay* analysis of the first part, a study of the end-to-end delay distribution in a mixed scenario with voice and data traffic and an admission control algorithm to provide voice traffic with end-to-end delay guarantees. With the proposed algorithm, voice packets are guaranteed an end-to-end delay below a certain threshold with a given probability, and therefore we can ensure that most of the packets will experience a low delay. While the first part of the paper is based on the earlier version of this paper in

☆ The work described in this paper is based on results of IST FP6 Integrated Project DAIDALOS. DAIDALOS receives research funding from the European Community's Sixth Framework Programme. Apart from this, the European Commission has no responsibility for the content of this paper.

* Corresponding author. Tel.: +34 91 624 8802; fax: +34 91 624 8849.
E-mail addresses: banchs@it.uc3m.es (A. Banchs), pablo@it.uc3m.es (P. Serrano), azcorra@it.uc3m.es (A. Azcorra).

[1] In case the packet is discarded, we consider its backoff delay equal to ∞.

0140-3664/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2005.08.006

[6], the second part is entirely novel and was not included in the short version.

The rest of the paper is structured as follows. In Section 2, we present a brief overview of the 802.11 DCF protocol. In Section 3, we propose a method to analyze the distribution of the saturation backoff delay. Simulation results show that, with our method, the probability that the delay falls below a certain value can be computed accurately and efficiently. In Section 4, we extend our analysis to study the end-to-end delay performance in a mixed scenario with voice and data traffic, and propose an algorithm to perform admission control in order to provide delay guarantees to voice packets. Simulation results show that our algorithm is effective in providing the committed delay guarantees while maximizing the number of admitted stations. Finally, in Section 5 we present our concluding remarks.

2. DCF

The DCF access method of the IEEE 802.11 standard [7] is based on the CSMA/CA protocol. A station with a packet to transmit senses the channel and, if it remains free for a DIFS time, it transmits. If the channel is sensed busy, the station waits until the channel becomes idle for a DIFS time, after which it starts a backoff process.

Upon starting the backoff process, the station generates a random value chosen from a uniform distribution in the range (0, CW-1), and initializes its backoff time counter with this value. The CW value is called Contention Window, and depends on the number of transmissions failed for the packet. At the first transmission attempt, CW is set equal to a value CW_{min}, and it is doubled after each unsuccessful transmission, up to a maximum value of CW_{max}. The CW_{min} and CW_{max} values are fixed by the standard.

The backoff time counter is decremented once every time interval T_e for which the channel is detected empty, 'frozen' when a transmission is detected on the channel, and reactivated when the channel is sensed empty again for a DIFS time (if the transmission is detected as successful[2]) or an EIFS time (if it is detected as unsuccessful). The station transmits when the backoff time counter reaches zero.

If the packet is correctly received, the receiving station sends an ACK frame after a SIFS time. If the ACK frame is not received within an ACK timeout time, a collision is assumed to have occurred and the packet transmission is rescheduled according to the given backoff rules. If the number of retransmissions reaches a predefined Retry Limit, the packet is discarded. Upon completing the backoff process (either with a success or with a discard), the transmitting station resets the CW to its initial value and starts a new backoff process; before this ends, a new packet cannot be transmitted.

The use of the Request to Send (RTS)/Clear to Send (CTS) mechanism is optional in 802.11. When this option is applied, upon the backoff counter reaching zero, the transmitting station sends an RTS frame to the receiving station, which responds with a CTS frame after a SIFS. The packet is then sent at a SIFS time after receiving the CTS, and it is followed by an ACK frame. The RTS/CTS mechanism serves the purpose of avoiding collisions in the case of hidden station, since the hidden station, just by detecting one frame among the RTS and CTS, can suitably delay its transmission and thus avoid the collision. In addition, this mechanism also improves the performance of the WLAN, since when it is used collisions occur with the short RTS frames instead of long data packets.

Another optional mechanism with DCF is the bursting mechanism. When using this mechanism, a station that gets access to the channel retains the right to access the channel after transmitting a successful packet and can transmit two or more consecutive packets. In this case, each packet is acknowledged with an individual ACK frame, and the next packet is transmitted after a SIFS time following the ACK frame.

3. Saturation backoff delay analysis

In this section we propose two alternative models to compute the distribution of the saturation backoff delay. The first model, which we refer to as *accurate analysis*, accurately computes the saturation backoff delay distribution. The second model, which we refer to as *simplified analysis*, provides a lower degree of accuracy but at a greatly reduced computational cost. The simplified analysis is used in Section 4 for the operations that require a high computational efficiency. For both models, we provide a basic analysis for fixed packet lengths and no RTS/CTS, and extensions for variable packet lengths and RTS/CTS.

3.1. Accurate analysis

Let us consider a WLAN with N stations operating under saturation conditions and sending packets of a fixed length l. Our objective is to compute the probability that, under these conditions, a packet transmission of a tagged station experiences a saturation backoff delay smaller than a given value D. We denote this probability by $P(d<D)$.

Fig. 1 shows the different components of the saturation backoff delay. Applying the theorem of the total probability, $P(d<D)$ can be decomposed as follows,

[2] We refer to a transmission being detected as successful if the received packet is error-free, and as unsuccessful if it contains errors. All stations check every received packet (regardless of its destination) for errors, using the Frame Check Sequence (FCS) field of the packet.

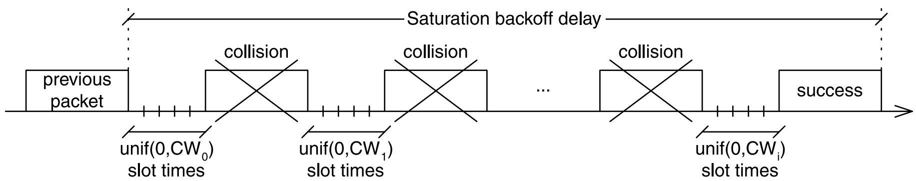


Fig. 1. Saturation backoff delay.

$$P(d<D)=\sum_{i=0}^{R}P(d<D/i\ \mathrm{col})P(i\ \mathrm{col}), \tag{1}$$

where $P(i$ col) represents the probability that a packet suffers i collisions before being successfully transmitted and R is the Retry Limit.

Let us define a slot time as the time interval between two consecutive backoff time decrements of the tagged station. Note that, according to this definition, a slot time may be either empty or contain the transmission of one or more stations. Applying to the previous equation the theorem of the total probability on the total number of slot times the tagged station counts down before transmitting successfully, we have

$$P(d<D)=\sum_{i=0}^{R}\sum_{j=0}^{W_i}P(d<D/i\,\mathrm{col},j\,\mathrm{slots})P(j\,\mathrm{slots}/i\,\mathrm{col})P(i\,\mathrm{col}), \tag{2}$$

where $W_i=\sum_{k=0}^{i}\mathrm{CW}_k-1$, with $\mathrm{CW}_k=\min(2^k\mathrm{CW}_{\min},\mathrm{CW}_{\max})$, and $P(j$ slots/i col) is the probability that the sum of the $i+1$ backoff times of the packet equals j,

$$P(j\ \mathrm{slots}/i\ \mathrm{col})=P\left(\sum_{k=0}^{i}\mathrm{unif}(0,\mathrm{CW}_k-1)=j\right), \tag{3}$$

where unif(0,C) represents a discrete random variable uniformly distributed in the range $\{0, 1,\ldots,C\}$.

As the probability mass function (pmf) of a sum of discrete random variables is equal to the convolution of the individual pmf's, we can compute $P(j$ slots/i col) as follows,

$$P(j\ \mathrm{slots}/i\ \mathrm{col})=(f_0*f_1*\ldots*f_i)_j, \tag{4}$$

f_k being the pmf of unif(0, CW_{k-1}). We compute the above convolution using Fast Fourier Transforms (FFT's).

Let τ be the probability that a station transmits in a slot time in a WLAN with N stations under saturation conditions. Note that this corresponds to the probability that the station's backoff counter reaches 0. Following the analysis of [8], which models the backoff process of a station with a Markov chain and computes the probability that a station transmits by adding all the state probabilities corresponding to a backoff counter of 0, τ can be expressed as

$$\tau=\frac{2(1-2p)(1-p^{R+1})}{W(1-(2p)^{m+1})(1-p)+(1-2p)(1-p^{R+1})+W2^m p^{m+1}(1-2p)(1-p^{R-m})}, \tag{5}$$

where R is the Retry Limit, p is the probability that a transmission attempt collides, $W=\mathrm{CW}_{\min}$ and m is such that $\mathrm{CW}_{\max}=2^m\mathrm{CW}_{\min}$.

The only unknown value from the above equation is the probability p that a transmission attempt collides. This corresponds to the probability that none of the other $N-1$ stations transmits in the slot time. Since all stations transmit with probability τ, this yields

$$p=1-(1-\tau)^{N-1}. \tag{6}$$

Eqs. (5) and (6) form a system of a non-linear equation which can be solved using numerical techniques and thus compute τ.

The first approximation upon which we base our analysis is the same as [9]: we assume that a station other than the tagged one transmits at each slot time with a constant and independent probability τ. With this assumption, the probability that the tagged station suffers i collisions before transmitting successfully can be computed from

$$P(i\ \mathrm{col})=p^i(1-p)=(1-(1-\tau)^{N-1})^i(1-\tau)^{N-1}. \tag{7}$$

Our second approximation is to assume that the saturation delay given i collisions and j slot times is a gaussian random variable[3], which we denote by d_{ij}. Note that, assuming independence between different slot times (which is given by the first approximation) and a large enough number of slot times (which is the typical case), the Central Limit Theorem assures that this approximation is accurate.

Given the above assumption, it is enough to know the average and the standard deviation of d_{ij} (which we denote by m_{ij} and σ_{ij}, respectively) to compute $P(d<D/i$ col, j

[3] This approximation is the key difference between our model and the analyses of [4,5]. While, with our approximation, we only need to compute the average and standard deviation values of d_{ij}, which can be performed efficiently, [4,5] compute all the possible values of d_{ij} and their probability, which has a high computational cost as d_{ij} can take a very large number of different values.

slots),

$$P(d < D/i\ \mathrm{col}, j\ \mathrm{slots}) = \begin{cases} 0.5 + 0.5\ \mathrm{erf}\left(\dfrac{D-m_{ij}}{\sqrt{2}\sigma_{ij}}\right), & \dfrac{D-m_{ij}}{\sigma_{ij}} \geq 0, \\ 0.5\ \mathrm{erfc}\left(-\dfrac{D-m_{ij}}{\sqrt{2}\sigma_{ij}}\right), & \dfrac{D-m_{ij}}{\sigma_{ij}} < 0. \end{cases} \tag{8}$$

Given the assumption of independence between different slot times, m_{ij} can be computed as the sum of the average duration of all slot times in d_{ij},

$$m_{ij} = jm_n + iT_c + T_s, \tag{9}$$

where m_n is the average duration of a slot time in which the tagged station does not transmit, T_c is the duration of a slot time that contains a collision and T_s is the duration of a slot time that contains a successful transmission.

The duration of a slot time that contains a successful transmission is equal to [10]

$$T_s = T_{\mathrm{PLCP}} + \frac{H+l}{C} + \mathrm{SIFS} + T_{\mathrm{PLCP}} + \frac{\mathrm{ACK}}{C} + \mathrm{DIFS}, \tag{10}$$

where T_{PLCP} is the PLCP (Physical Layer Convergence Protocol) preamble and header transmission time, H is the MAC overhead (header and FCS), ACK is the length of an ACK frame and C is the channel bit rate.

Similarly, the duration of a slot time that contains a collision is equal to

$$T_c = T_{\mathrm{PLCP}} + \frac{H+l}{C} + \mathrm{EIFS}. \tag{11}$$

The average duration of a slot time in which the tagged station does not transmit, m_n, is computed as

$$m_n = P_{s,n}T_s + P_{c,n}T_c + P_{e,n}T_e, \tag{12}$$

where $P_{s,n}$ represents the probability that a slot time in which the tagged station does not transmit contains a successful transmission, $P_{c,n}$ the probability that it contains a collision and $P_{e,n}$ the probability that it is empty. $P_{s,n}$, $P_{e,n}$ and $P_{c,n}$ can be computed from τ and N as follows,

$$P_{s,n} = (N-1)\tau(1-\tau)^{N-2}, \tag{13}$$

$$P_{e,n} = (1-\tau)^{N-1}, \tag{14}$$

and

$$P_{c,n} = 1 - P_{s,n} - P_{e,n}. \tag{15}$$

With the assumption of independence between different slot times, the standard deviation σ_{ij} can be computed from

$$\sigma_{ij}^2 = j\sigma_n^2, \tag{16}$$

where σ_n is the standard deviation of the duration of a slot time in which the tagged station does not transmit,

$$\sigma_n^2 = P_{s,n}T_s^2 + P_{c,n}T_c^2 + P_{e,n}T_e^2 - m_n^2. \tag{17}$$

3.2. RTS/CTS

In case the RTS/CTS option is used, successful packet transmissions are preceded by an RTS/CTS exchange, while collisions occur with RTS frames instead of data packets. Accordingly, the durations of the slot times containing a successful transmission and a collision are computed as [10]

$$T_s = T_{\mathrm{PLCP}} + \frac{\mathrm{RTS}}{C} + \mathrm{SIFS} + T_{\mathrm{PLCP}} + \frac{\mathrm{CTS}}{C} + \mathrm{SIFS} + T_{\mathrm{PLCP}} + \frac{H+l}{C} + \mathrm{SIFS} + T_{\mathrm{PLCP}} + \frac{\mathrm{ACK}}{C} + \mathrm{DIFS} \tag{18}$$

and

$$T_c = T_{\mathrm{PLCP}} + \frac{\mathrm{RTS}}{C} + \mathrm{EIFS}, \tag{19}$$

where RTS and CTS are the length of the RTS and CTS frames.

With the only modification of taking the above expressions to compute T_s and T_c, the analysis in the previous clause can be used to compute the saturation backoff delay distribution for the RTS/CTS case.

3.3. Variable packet lengths

Next, we extend our model to the case when packet lengths are not fixed but follow a certain distribution. Specifically, we consider that a packet length takes a value l of the set L with probability P_l, L being the set of all possible packet lengths. For simplicity, we assume that all stations transmit the same packet length distribution; however, the analysis would be very similar in the case when this condition does not hold.

In order to account for variable packet lengths, we have to modify the expressions to obtain the m_{ij} and σ_{ij} values. m_{ij} is computed as

$$m_{ij} = jm_n + im_c + m_s, \tag{20}$$

where m_c is the average duration of a slot time in which the tagged station collides and m_s is the average duration of a slot time in which the tagged station transmits a packet successfully.

The average duration of a slot time in which the tagged station does not transmit, m_n, is computed as

$$m_n = \sum_{l\in L} P_{s,l,n}T_{s,l} + \sum_{l\in L} P_{c,l,n}T_{c,l} + P_{e,n}T_e \tag{21}$$

where $P_{s,l,n}$ represents the probability that a slot time in which the tagged station does not transmit contains a successful transmission of a packet of length l, $P_{c,l,n}$

the probability that it contains a collision with the longest packet involved being of length l, and $T_{s,l}$ and $T_{c,l}$ are the slot time durations in each case.

$P_{s,l,n}$ and $P_{c,l,n}$ are obtained from

$$P_{s,l,n} = (N-1)\tau(1-\tau)^{N-2}P_l, \tag{22}$$

$$P_{c,l,n} = (1-P_{s,l,n}-P_{e,n})P_{c,l}, \tag{23}$$

where $P_{c,l}$ is the probability that the longest packet involved in a collision is of length l. Neglecting the collisions of more than two stations,

$$P_{c,l} = 2P_l\sum_{k\in L_l}P_k - P_l^2, \tag{24}$$

where L_l is the set of all the packet lengths smaller than or equal to l.

The duration of a slot time that contains a successful transmission of a packet of length l, $T_{s,l}$, and the duration of a slot time that contains a collision with the longest packet involved of length l, $T_{c,l}$, can be calculated as a function of l from Eqs. (10) and (11).

m_s and m_c are calculated as follows,

$$m_s = \sum_{l\in L}P_l T_{s,l}, \tag{25}$$

$$m_c = \sum_{l\in L}P_{c,l}T_{c,l}. \tag{26}$$

Finally, the standard deviation σ_{ij} for the variable packet length case can be computed from

$$\sigma_{ij}^2 = j\sigma_n^2 + i\sigma_c^2 + \sigma_s^2 \tag{27}$$

with

$$\sigma_n^2 = \sum_{l\in L}P_{s,l,n}T_{s,l}^2 + \sum_{l\in L}P_{c,l,n}T_{c,l}^2 + P_{e,n}T_e^2 - m_n^2. \tag{28}$$

Note that in the above analysis we have assumed that the RTS/CTS option is not used. However, the model can be easily extended to the RTS/CTS case, simply by following the analysis presented in this section but using the expressions for $T_{s,l}$ and $T_{c,l}$ given by Eqs. (18) and (19) as a function of l.

3.4. Simplified analysis

One of the drawbacks of the analysis provided in the previous sections to compute $P(d<D)$ is that it requires performing a large number of *erf* or *erfc* operations, which has a non-negligible computational cost. We now present an alternative analysis that, although it gives less accurate results, it is more computationally efficient.

In our simplified analysis, to compute $P(d<D)$ we apply the theorem of the total probability as follows,

$$P(d<D) = \sum_{j=0}^{W_{max}}P(d<D/j \text{ slots})P(j \text{ slots}) \tag{29}$$

where j is the total number of slot times elapsed between the beginning of the backoff process of a packet and its successful transmission (including the slot times in which the tagged station transmits),

$$P(j \text{ slots}) = \sum_{i=0}^{R}P(j \text{ slots}/i \text{ col})P(i \text{ col}) \tag{30}$$

with

$$P(j \text{ slots}/i \text{ col}) = P\left(\sum_{k=0}^{i}\text{unif}(1,\text{CW}_k) = j\right) \tag{31}$$

and $W_{max} = \sum_{k=0}^{R}\text{CW}_k$.

The key approximation of our simplified analysis is to assume that all slot times (regardless of whether the tagged station transmits or not) have a fixed duration equal to the average duration of a slot time. With this assumption,

$$P(d<D/j \text{ slots}) = \begin{cases} 1, & D < jT_{slot}, \\ 0, & D \geq jT_{slot}, \end{cases} \tag{32}$$

where T_{slot} is the average duration of a slot time,

$$T_{slot} = P_sT_s + P_cT_c + P_eT_e, \tag{33}$$

where P_s represents the probability that a slot time contains a successful transmission, P_c the probability that it contains a collision and P_e the probability that it is empty, and T_s, T_c and T_e are the corresponding average slot time durations,

$$P_s = N\tau(1-\tau)^{N-1}, \tag{34}$$

$$P_e = (1-\tau)^N, \tag{35}$$

$$P_c = 1 - P_s - P_e. \tag{36}$$

Note that this simplified analysis can easily be extended to the RTS/CTS and variable packet length cases by using the expressions for T_s and T_c given in Sections 3.2 and 3.3.

3.5. Performance evaluation

Next, we evaluate the accuracy and computational efficiency of the proposed model. The values of the system parameters used to obtain the results have been taken from the 802.11b physical layer [11]. The packet length is equal to 1500 bytes for the fixed packet length case, and has been derived from the measurements of Internet traffic presented in [12] for the variable packet length case. Simulations are performed with the 802.11 DCF simulator used in [6]. This is an event-driven simulator that closely follows

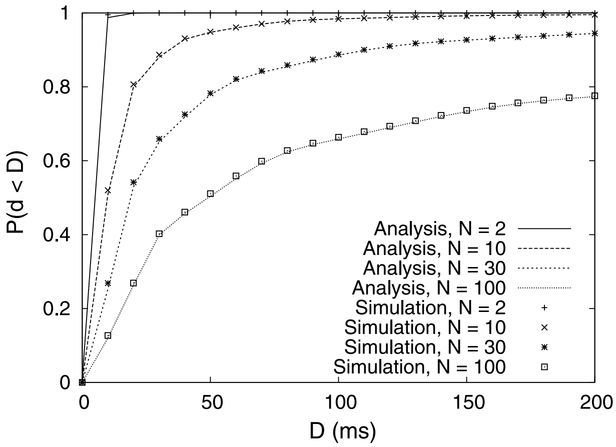


Fig. 2. Saturation backoff delay cdf: accurate model.

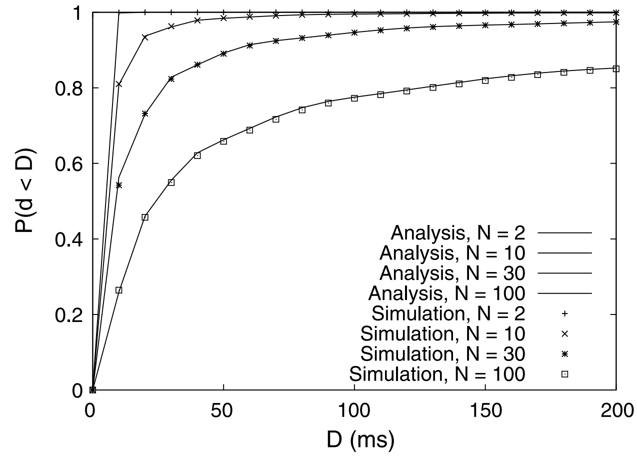


Fig. 4. Saturation backoff delay cdf: variable packet lengths extension.

the details of the MAC protocol of 802.11 DCF for each independently transmitting station.

Figs. 2, 3 and 4 show the cumulative distribution function (cdf) of the saturation delay, for our accurate model, RTS/CTS extension and variable packet lengths extension, respectively. Analytical results are represented with lines and simulations with points. Simulation results are given with a 95% confidence interval below 0.1%. Results show that our analysis is accurate; in all cases, and for all values of D and N, simulations results closely follow the analytical ones.

In order to evaluate the accuracy of our simplified analysis as compared to the accurate analysis, we repeated the experiment of Fig. 2 for both analyses; results are given in Fig. 5. We observe from the results that, for low delays, the accurate analysis closely follows simulations results, while the simplified analysis diverges significantly at some points. In particular, with the accurate analysis errors do not exceed 1%, while with the simplified analysis there are error peaks of about 10%. As delay increases, both analyses become accurate. Notice that, for the low delay values represented in the graph, simulation results oscillate. The reason is that the duration of the non-empty slot times is not negligible as compared to the considered delay values.

In order to evaluate the computational efficiency of the two methods proposed (the accurate and the simplified analyses), we measured the times required to compute the cdf values. Table 1 gives the times required to compute $P(d<D)$ for different values of N and D with the two methods. Measurements have been taken in a Pentium 4 PC with 2.66 GHz of CPU speed and 192 MB of RAM, running under the Linux operating system with the Mandrake 10.0 standard installation and the Linux kernel version 2.6. It can be seen from the results that the simplified analysis is computationally much more efficient. Specifically, the computational times for the simplified analysis (of several tenths of μs) are three orders of magnitude smaller than the ones for the accurate analysis (of several tenths of ms).

We also observe from the results of Table 1 that the computational times for the accurate analysis are fairly independent of D and N, while for the simplified analysis computational times increase with D and decrease with N.

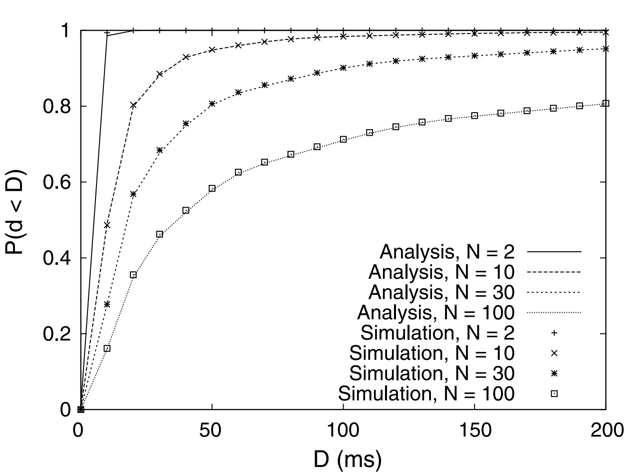


Fig. 3. Saturation backoff delay cdf: RTS/CTS extension.

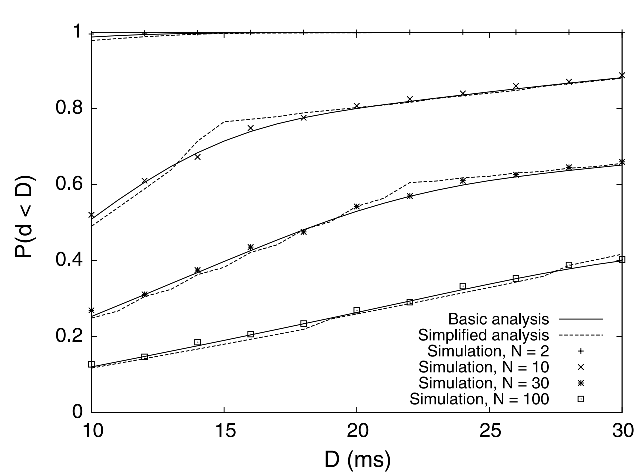


Fig. 5. Accurate vs. simplified analysis.

Table 1
Computational times (in ms)

	D (ms)	$N=2$	$N=10$	$N=30$	$N=100$
Accurate	20	68	70	66	58
	100	66	71	66	57
	200	57	68	62	55
Simplified	20	0.03	0.03	0.02	0.02
	100	0.06	0.04	0.03	0.03
	200	0.14	0.09	0.04	0.04

The reason is that for the accurate analysis we have to compute the contribution of all the possible combinations of i and j in Eq. (2), and the number of possible combinations is independent of D and N. In contrast, for the simplified analysis we only have to account for the values of j in Eq. (29) whose delay is smaller than D according to Eq. (32), and the number of j values that meet this condition increases with D and decreases with N.

We conclude from the results presented in this section that the accurate analysis and its derivatives (RTS/CTS and variable packet lengths[4]) provide more accurate results at the price of a higher computational cost, while the simplified analysis is less accurate for low delay values but its computational cost is minimal.

4. End-to-end delay guarantees

In this section, we present an analytical model for the end-to-end delay distribution of a WLAN and, based on this model, an admission control algorithm that provides real-time applications with end-to-end delay guarantees. By end-to-end delay in this context we mean the time elapsed since a packet is generated by an application until it is transmitted successfully in the WLAN.

The analysis provided here is based on the methods to compute the distribution of the backoff delay presented in Section 3. Specifically, in the analysis that follows we first use the simplified analysis for the resolution of a non-liner equation, and later we use the accurate analysis for an operation that is performed only once. Using the simplified analysis in the first case avoids an unacceptably high computational cost resulting from the large number of iterations required to solve the non-linear equation. In the second case, since the operation is performed only once, the accurate analysis can be used without paying a high price in terms of computational cost.

Hereafter, we refer with $A_{\mathrm{cdf}}(D)$ to the probability $P(d<D)$ computed according to the accurate analysis (and its derivatives for the RTS/CTS and variable packet length cases) of Sections 3.1, 3.2 and 3.3, and with $S_{\mathrm{cdf}}(D)$ to the probability $P(d<D)$ computed according to the simplified analysis of Section 3.4.

4.1. Model

Although the analysis we present here can be applied to a more general framework, in the rest of this paper we have decided to focus on a specific scenario which we believe represents a realistic environment for a WLAN. In our model, we assume the following configuration of the 802.11 DCF stations:

- WLAN stations transmit either voice or data traffic (hereafter, we refer to the former as 'voice stations' and to the latter as 'data stations').
- Voice stations generate packets according to the widely used voice codec of [13]. This codec generates 260 bits of application data payload at a rate of 13 Kbps, to which the RTP/UDP/IPv6 overhead is added.
- When a voice station gains access to the channel, it uses the bursting mechanism of DCF to transmit all the enqueued voice packets awaiting for transmission[5]. Note that the use of this mechanism improves the delay performance of voice packets, as it avoids executing an independent backoff process for each packet.
- Data stations follow the behavior described in [14] for elastic applications. Specifically, they always have packets ready for transmission (note that this is the typical behavior of widely used data applications like e.g. ftp or http downloads). The data packets length follows the traffic measurements of [12].
- The RTS/CTS mechanism is used both by voice and data stations. Indeed, the results of Section 3.5 show that the performance of the WLAN increases when this mechanism is used, as a result of avoiding collisions of long packets.

4.2. End-to-end delay analysis for voice applications

Let us consider a WLAN with N_{v} voice stations and N_{d} data stations configured according to the above model. Our objective is to compute the distribution of the delay elapsed between the generation of a voice packet and its successful transmission.

Let τ_{v} denote the probability that a given voice station transmits in a randomly chosen slot time and p_{v} the probability that a transmission of a voice station collides. Let τ_{d} and p_{d} denote the same probabilities for a data station.

[4] Results on the computational efficiency of the accurate analysis for the RTS/CTS and the variable packet length cases are given in [6]. These results show that the computational efficiency for these cases is similar to the efficiency of the fixed packet length model.

[5] We note that, although the bursting mechanism of 802.11 DCF was originally designed to transmit fragments of the same packet, it is currently being used in some WLAN products for transmitting bursts of different packets in order to improve performance. See e.g. the *frame-bursting technology* of Broadcom products (http://www.broadcom.com).

Our analysis is outlined as follows. We focus on the variable p_d and present two alternative ways of computing the average number of voice packets transmitted in a burst as a function of p_d, which we denote respectively by $n(p_d)$ and $n'(p_d)$. Then, by solving the non-linear equation $n(p_d)=n'(p_d)$ we obtain p_d, from which the rest of the analysis follows.

Let us start by assuming that the probability p_d is known. Then, since data stations are saturated, τ_d can be computed from p_d with the formula given in [8] for saturated stations,

$$\tau_d = \frac{2(1-2p_d)\left(1-p_d^{R+1}\right)}{W(1-(2p_d)^{m+1})(1-p_d)+(1-2p_d)\left(1-p_d^{R+1}\right)+W2^m p_d^{m+1}(1-2p_d)\left(1-p_d^{R-m}\right)}. \tag{37}$$

With the assumption that the transmission probabilities are independent, the following equation holds for p_d,

$$p_d = 1-(1-\tau_d)^{N_d-1}(1-\tau_v)^{N_v}, \tag{38}$$

from which τ_v and p_v can be obtained,

$$\tau_v = 1-\sqrt[N_v]{\frac{1-p_d}{(1-\tau_d)^{N_d-1}}}, \tag{39}$$

$$p_v = 1-(1-\tau_v)^{N_v-1}(1-\tau_d)^{N_d}. \tag{40}$$

The average duration of a slot time can be computed from

$$T_{slot} = P_e T_e + P_{s,v} T_v + P_{s,d} T_d + P_c T_c, \tag{41}$$

where P_e, $P_{s,v}$, $P_{s,d}$ and P_c are the probabilities that a slot time is empty, contains a successful transmission of a voice station, a success of a data station or a collision, respectively, and T_e, $T_{s,v}$, $T_{s,d}$ and T_c are the corresponding average slot time durations.

P_e, $P_{s,v}$, $P_{s,d}$ and P_c are computed according to

$$P_e = (1-\tau_d)^{N_d}(1-\tau_v)^{N_v}, \tag{42}$$

$$P_{s,v} = N_v\tau_v(1-\tau_v)^{N_v-1}(1-\tau_d)^{N_d}, \tag{43}$$

$$P_{s,d} = N_d\tau_d(1-\tau_d)^{N_d-1}(1-\tau_v)^{N_v}, \tag{44}$$

$$P_c = 1-P_e-P_{s,v}-P_{s,d}. \tag{45}$$

$T_{s,d}$ and T_c are computed according to Section 3. $T_{s,v}$ is obtained from

$$T_{s,v} = T_{PLCP} + \frac{RTS}{C} + SIFS + T_{PLCP} + \frac{CTS}{C} + n\left(SIFS + T_{PLCP} + \frac{H+l_v}{C} + SIFS + T_{PLCP} + \frac{ACK}{C}\right) + DIFS, \tag{46}$$

where l_v is the payload size of a voice packet and n is the average number of packets that a voice station transmits when it gets access to the channel.

With the assumption that no voice packets are dropped[6], the aggregated data rate generated by all the voice stations is equal to the aggregated rate of successfully transmitted voice data in the WLAN, and therefore the following equation holds,

$$\frac{N_v l_v}{T} = \frac{P_{s,v} n l_v}{T_{slot}}, \tag{47}$$

where T is the interval between two packet arrivals in a voice station.

Eqs. (41)–(47) form a first order equation from which we can isolate n. As a result, starting from p_d we can compute the value of n. We denote by $n(p_d)$ this procedure to compute n.

We now present the alternative procedure to obtain n from p_d, denoted by $n'(p_d)$. Given p_v and T_{slot} computed as above, the cdf of the backoff delay can be easily calculated following Section 3.4. From this, the probability that the backoff delay falls in the interval $(iT, (i+1)T)$ can be obtained as follows,

$$P(iT < d < (i+1)T) = S_{cdf}((i+1)T) - S_{cdf}(iT). \tag{48}$$

Since, with the bursting mechanism, the backoff process always starts with a new packet, after a backoff process duration of $(iT, (i+1)T)$ a station will have $i+1$ voice packets to transmit, hence the probability that a transmission of a voice station contains a burst of j packets is equal to

$$P_j = P((j-1)T < d < jT). \tag{49}$$

From the above, the average number of packets that a voice station transmits when it get access to the channel can be computed as

$$n'(p_s) = \sum_{j=0}^{\max}(j+1)P_{j+1}, \quad \text{where} \quad \max = \sum\nolimits_{k=0}^{R} CW_k T_{slot}/T. \tag{50}$$

The above results in the following non-linear equation on p_d, that can be solved using numerical techniques,

$$n'(p_d) = n(p_d), \tag{51}$$

[6] Notice that this assumption is accurate as long as voice traffic is served with reasonable quality. For our objective of performing admission control, we are interested in the WLAN configurations that satisfy this condition.

from which we can obtain p_d. Specifically, we solve the above non-linear equation using the bisection method. Note that, as a result of using our simplified analysis for the computation of the cdf in Eq. (48), the computational cost of resolving the non-liner equation is reasonably low, and does not grow unacceptably due to the large number of iterations required to solve the non-linear equation.

Once we have solved the non-linear equation, we proceed as follows. From p_d, we compute $P_{s,v}$, $P_{s,d}$, p_v and P_j for $j \in \{1,\ldots,\max\}$ with the equations given above. Once these values are known, the scenario that we have can be interpreted simply as one with variable packet lengths, which has been analyzed in Section 3.3.

With the above interpretation, we can calculate the cdf of the backoff delay following Section 3.3. Note that, in contrast to Eq. (48), now we do not use the simplified analysis but the accurate one. The reason is that the computation does not need to be performed iteratively but only once, and therefore a more expensive analysis can be used to gain in accuracy without paying a high price in terms of computational time.

The remaining challenge is to compute, from the cdf of the backoff delay, the end-to-end delay distribution. The probability that a transmission of a voice station is successful and contains a burst of i packets can be computed as

$$P_i = A_{\mathrm{cdf}}(iT) - A_{\mathrm{cdf}}((i-1)T). \tag{52}$$

Similarly, we can calculate the probability that a backoff process of a voice station, either leading to a successful transmission or a discard due to reaching the retry limit, contains i packets,

$$P'_i = A'_{\mathrm{cdf}}(iT) - A'_{\mathrm{cdf}}((i-1)T), \tag{53}$$

where $A'_{\mathrm{cdf}}(D)$ is computed exactly as in Section 3.3 with the only difference that in Eq. (1) we also account for the case of $R+1$ collisions, i.e. the case of a burst of packets being discarded after reaching the retry limit.

With the above, we can obtain the probability that a randomly chosen packet is transmitted successfully in a transmission with a burst of i packets,

$$P_{s,i} = \frac{iP_i}{\sum_{j=1}^{\max} jP'_j}, \tag{54}$$

from which the probability that a randomly chosen packet is the jth packet of a successful transmission of a burst of i packets can be computed easily,

$$P_{s,i,j} = \frac{1}{j} P_{s,i}. \tag{55}$$

Finally, the probability that a voice packet suffers an end-to-end delay smaller than D can be calculated as follows,

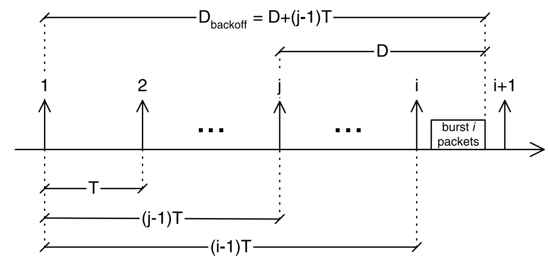


Fig. 6. Delay of the jth packet of a burst of i packets.

$$P(d_v < D) = \sum_{i=1}^{\max} \sum_{j=1}^{i} P(d_v < D/i,j) P_{s,i,j}, \tag{56}$$

where $P(d_v < D/i, j)$ is the probability that the delay of the jth packet of a successful transmission of a burst of i packets is smaller than D,

$$P(d_v < D/i,j) = \begin{cases} 1, & D > (i-j+1)T, \\ \hat{P}, & (i-j)T < D < (i-j+1)T, \\ 0, & D < (i-j)T, \end{cases} \tag{57}$$

with (Fig. 6)

$$\hat{P} = \frac{A_{\mathrm{cdf}}(D + (j-1)T) - A_{\mathrm{cdf}}((i-1)T)}{A_{\mathrm{cdf}}(iT) - A_{\mathrm{cdf}}((i-1)T)}. \tag{58}$$

4.3. Admission control algorithm

It is well-known that real-time applications require that most of their packets reach the destination with a delay lower than a given threshold, or otherwise they see their performance sharply degraded. Following this, in this section we propose, based on the analysis of the previous section, an admission control algorithm that guarantees to voice packets a delay below a certain threshold D with a given probability P, where P and D are input parameters to the algorithm.

Assume that the WLAN is operating with a number of voice and data stations N_v and N_d, such that the above requirement for voice traffic is met. When a new voice station issues a request for admission, we proceed as follows. We compute, using the analysis given in Section 4.2, the probability that a voice packet suffers a delay below D given N_v+1 voice stations and N_d data stations. If this probability is higher than P, this means that the requirement is met with the new configuration and the voice station can be admitted. If it is smaller, the admission request has to be rejected in order to preserve the desired quality for voice. If the new station requesting admission is a data station, we proceed in the same way.

Note that the above delay guarantees refer only to the WLAN part of the network (according to our definition of end-to-end delay). If the end-to-end communication path

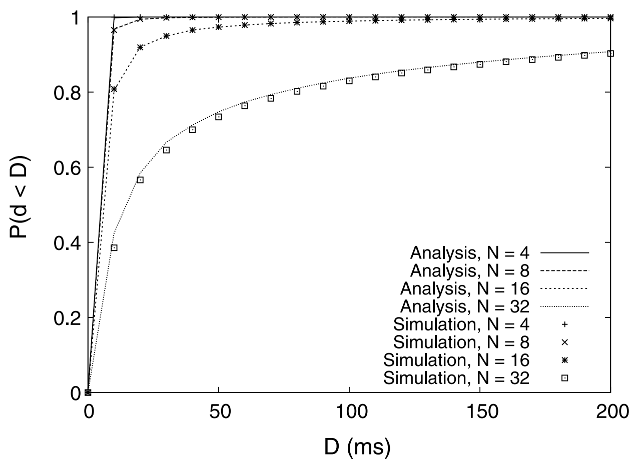


Fig. 7. End-to-end delay cdf: voice scenario.

involves other parts besides the WLAN, the guarantees provided here would only account for some portion of the total delay and should be added to the other portions [15]. This is out of the scope of this paper.

4.4. Performance evaluation

In this section, we evaluate the accuracy and efficiency of the end-to-end delay analysis and the proposed admission control algorithm. We consider three scenarios: *voice scenario*, in which 75% of the stations are voice stations and the remaining 25% are data stations, *mixed scenario*, in which 50% of the stations are voice and the other 50% data, and *data scenario*, with 75% of data stations and 25% of voice stations. Unless otherwise specified, we use the same parameters as in Section 3.5.

Figs. 7–9 show the cumulative distribution function (cdf) of the end-to-end delay for voice packets, for the voice scenario, mixed scenario and data scenario, respectively. Analytical results are represented with lines and simulations with points. Results show that our analysis is accurate; in all

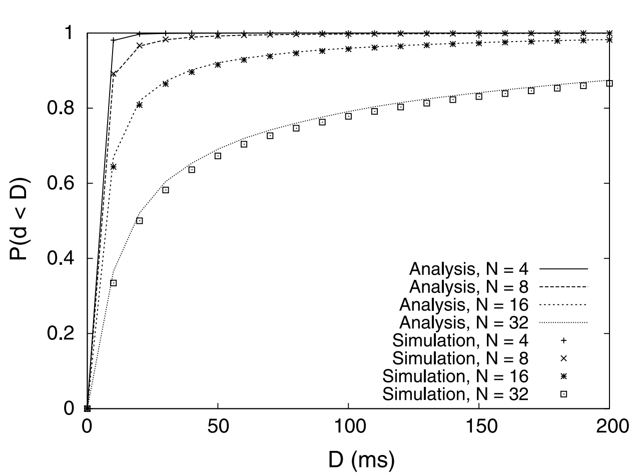


Fig. 8. End-to-end delay cdf: mixed scenario.

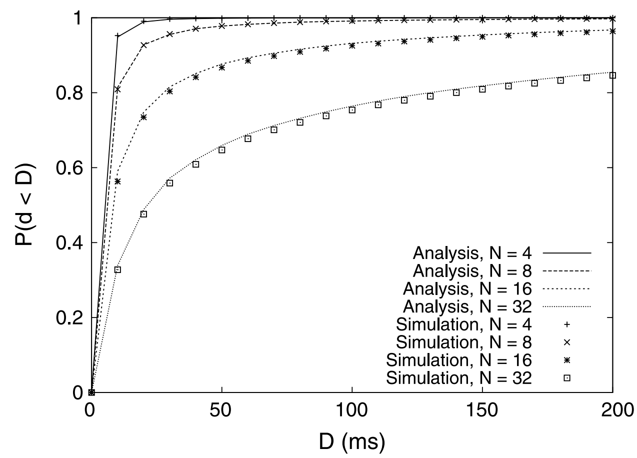


Fig. 9. End-to-end delay cdf: data scenario.

cases, the difference between simulation and analytical results does not exceed 1%.

We observe from the figures that there is some decrease in delay performance as the proportion of data stations increases (e.g. the probability that the delay is smaller than 200 ms for $N=32$ is equal to 90% for the voice scenario, 87% for the mixed scenario and 85% for the data scenario). The reason is that data stations transmit more often (as they are saturated) as well as longer packets [12].

In order to assess the effectiveness of the proposed admission control algorithm, we proceed as follows. For the three scenarios considered (voice, data and mixed), we evaluate the maximum number stations that can be accepted while preserving the desired voice quality for different values of D and P. Specifically, we evaluate the performance for an increasing number of stations (distributed between voice and data stations according to the given scenario) and choose the last one that meets the delay requirements. This is done both analytically and via simulation, and the maximum resulting number of stations in each case, N_{analysis} and $N_{\text{simulation}}$, is given in Tables 2–4. For each $\{P, D\}$ duple, we give the $P(d<D)$ obtained via simulation with $N=N_{\text{analysis}}$.

It can be seen from [16] that, for the considered voice codec, drop rates below 1% lead to a high voice quality, with 5% the quality is acceptable and with 10% it is poor.

Table 2
Maximum number of stations: voice scenario

D (ms)	P	N_{analysis}	$N_{\text{simulation}}$	$P(d<D)\vert_{N=N_{\text{analysis}}}$
20	0.9	16	16	0.919
	0.95	12	12	0.973
	0.99	8	8	0.994
50	0.9	20	20	0.931
	0.95	16	16	0.973
	0.99	12	12	0.994
150	0.9	28	28	0.922
	0.95	24	24	0.952
	0.99	16	16	0.994

Table 3
Maximum number of stations: mixed scenario

D (ms)	P	N_{analysis}	$N_{\text{simulation}}$	$P(d<D)\vert_{N=N_{\text{analysis}}}$
20	0.9	**12**	10	0.892
	0.95	8	8	0.967
	0.99	4	4	0.997
50	0.9	16	16	0.915
	0.95	12	12	0.962
	0.99	8	8	0.993
105	0.9	24	24	0.910
	0.95	18	18	0.958
	0.99	12	12	0.991

Table 4
Maximum number of stations: data scenario

D (ms)	P	N_{analysis}	$N_{\text{simulation}}$	$P(d<D)\vert_{N=N_{\text{analysis}}}$
20	0.9	8	8	0.927
	0.95	4	4	0.991
	0.99	**0**	4	–
50	0.9	12	12	0.930
	0.978	8	8	0.995
	0.99	4	4	0.999
150	0.9	20	20	0.917
	0.95	**16**	12	0.949
	0.99	8	8	0.996

According to [17], a total delay that does not exceed 150 ms is acceptable for voice applications. However, it needs to be taken into account that the WLAN in some cases may account only for a portion of the total delay. Based on these considerations, in our admission control we have considered the delay thresholds of $D=20$, 50, and 150 ms and the probabilities of $P=0.9$, 0.95 and 0.99.

The results obtained confirm the effectiveness of our admission control algorithm. Indeed, the maximum number of stations that can be admitted according to our algorithm (N_{analysis}) coincides with the maximum number of stations admissible according to simulation results in all cases but three (shown in bold). In two cases, our algorithm admits more stations. However, in both cases the simulated performance keeps very close to the target ($P=0.892$ vs. $P=0.9$ and $P=0.949$ vs. $P=0.95$). In the third case, our algorithm does not admit any station while four stations could be admitted[7]. The P obtained analytically in this case with four stations is 0.989 while the one given by simulations is 0.991, so this also represents a borderline case in which the delay requirements ($P=0.99$) are barely met with four stations. We conclude that our algorithm achieves, with a high degree of accuracy, the goal of admitting as many stations as possible while guaranteeing the committed performance.

We finally evaluate the computational efficiency of our admission control algorithm. Table 5 gives the times required to perform an admission decision for $N=N_{\text{analysis}}$, for the different scenarios and varying values of D and P. We can observe that in all cases, this time keeps below 0.25 s. The times obtained are fully acceptable for admission control, which confirms that our algorithm can be used to perform admission control decisions at run-time.

5. Summary and final remarks

As the capacity of WLANs and their use by realtime applications increases, there arises the need for better understanding and predicting the delay behavior of this type of networks. In particular, algorithms are necessary in order to provide voice applications, which are the most popular real-time applications used in this type of environment, with the required performance guarantees.

In the first part of this paper, we propose a method to compute the distribution of the backoff delay in 802.11 DCF under saturation conditions. Simulation and analytical results show that our method is accurate and computationally efficient. Our basic analysis of the backoff delay distribution assumes that all transmissions are of the same fixed packet length, but is later extended to account for variable packet lengths. None of the previous delay analyses of 802.11 DCF [1–5] accounts for non fixed packet lengths.

In the second part of the paper we analyze, based on the study of the first part, the end-to-end delay suffered by voice traffic in a mixed scenario with voice and data stations. In our configuration for voice traffic, we use the bursting and RTS/CTS mechanisms of 802.11 DCF, as these mechanisms contribute to reduce the delay experienced by voice stations. Simulation results confirm the accuracy of our analysis.

We then propose an admission control algorithm based on the end-to-end delay analysis. Simulations show that our admission control guarantees the desired performance to voice applications while maximizing the number of stations than can be admitted in the WLAN. The computational cost of the algorithm is sufficiently low to allow a run-time usage. Results show that, with the proposed algorithm, real-time applications (and specifically voice) can be effectively supported in 802.11 DCF WLAN.

There have been many protocol proposals for WLAN in the literature that, unlike DCF, have been designed

Table 5
Computational times (in ms)

D (ms)	P	Voice	Mixed	Data
20	0.9	80	82	76
	0.95	70	69	63
	0.99	65	71	–
50	0.9	161	168	166
	0.95	146	139	126
	0.99	128	103	81
150	0.9	223	203	206
	0.95	188	176	203
	0.99	129	138	116

[7] Note that 4 is the smallest number of stations that can be admitted in this case, in order to meet the requirement of the scenario that 75% of the stations are data stations and the other 25% are voice stations.

specifically to satisfy the delay requirements of real-time applications (e.g. [18–20]). The PCF scheme of 802.11 [7] was also designed with a similar intention. However, none of these (including PCF) is widely deployed today, which leaves DCF as the only practical option to provide real-time traffic communication in WLANs.

The IEEE 802.11 WG is currently undergoing a standardization activity to extend the 802.11 protocol with QoS support, leading to the upcoming 802.11e standard [21], which will provide a more efficient means for supporting real-time traffic. The EDCA access mechanism of 802.11e is an extension of the DCF protocol. We believe that our analysis here provides a basis for the study of the end-to-end delay distribution of 802.11e EDCA, in a similar way that our analyses of EDCA in [22,23] extended previous works of DCF [8,10]. The other access mechanism of 802.11e, HCCA, extends the PCF of 802.11. The deployment roadmap of HCCA is, at this point in time, less clear than EDCA.

References

[1] E. Ziouva, T. Antonkopoulos, CSMA/CA Performance under high traffic conditions: throughput and delay analysis, Computer Communications Journal 25 (1) (2002) 313–321.

[2] P. Chatzimisios, A.C. Boucouvalas, V. Vitsas, Packet delay analysis of IEEE 802.11 MAC protocol, IEE Electronics Letters 39 (18) (2003) 1358–1359.

[3] B. Li, R. Battiti, Performance Analysis of an Enhanced IEEE 802.11 Distributed Coordination Function Supporting Service Differentiation, in: Proceedings of QofIS'03, Stockholm, Sweden, October 2003.

[4] O. Tickoo, B. Sikdar, Queueing Analysis and Delay Mitigation in IEEE 802.11 Random Access MAC based Wireless Networks, in: Proceedings of IEEE INFOCOM'04, Hong Kong, China, March 2004.

[5] H. Zhai, Y. Fang, Performance of Wireless LANs Based on IEEE 802.11 MAC Protocols, in: Proceedings of IEEE PIMRC'03, September 2003.

[6] A. Banchs, Analysis of the backoff delay distribution in 802.11 DCF: a step towards end-to-end delay guarantees in WLANs, in: Proceedings of QofIS'04, Barcelona, Spain, September 2004.

[7] IEEE 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Standard, IEEE, August 1999.

[8] H. Wu, Y. Peng, K. Long, S. Cheng, J. Ma, Performance of Reliable Transport Protocol over IEEE 802.11 wireless LAN: analysis and enhancement, in: Proceedings of IEEE INFOCOM'02, New York City, NY, June 2003.

[9] F. Cali, M. Conti, E. Gregori, Dynamic Tuning of the IEEE 802.11 Protocol to achieve a theoretical throughput limit, IEEE/ACM Transactions on Networking 8 (6) (2000) 785–799.

[10] G. Bianchi, G. Bianchi, Performance analysis of the IEEE 802.11 Distributed Coordination Function, IEEE Journal on Selected Areas in Communications 18 (3) (2000) 535–547.

[11] IEEE 802.11b, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: high-speed physical layer extension in the 2.4 GHz Band, Supplement to IEEE 802.11 Standard, September 1999.

[12] K. Claffy, G. Miller, K. Thompson, The nature of the beast: Recent traffic measurements from an internet backbone, in: Proceedings of INET'98, Geneve, Switzerland, July 1998.

[13] European Telecommunications Standards Institute, European digital cellular telecommunications system (Phase 1); GSM Full Rate Speech Transcoding (GSM 06.10), January 1995.

[14] S. Shenker, Fundamental Design Issues for the Future Internet, IEEE Journal on Selected Areas in Communications 13 (7) (1995) 1176–1188.

[15] T.F. Znati, R. Melhem, Node delay assignment strategies to support end-to-end delay requirements in heterogeneous networks, IEEE/ACM Transactions on Networking 12 (5) (2004) 879–892.

[16] ITU-T, Provisionan plannig values for the equipment impairment factor I_e, Appendix to Recommendation G.113 (Revised Draft), September 1999.

[17] ITU-T, One-way transmission time, Recommendation G.114, February 1996.

[18] V. Kanodia, C. Li, B. Sadeghi, A. Sabharwal, E. Knightly, Distributed multi-hop with delay and throughput constraints, in: Proceedings of MOBICOM'01, Rome, Italy, July 2001.

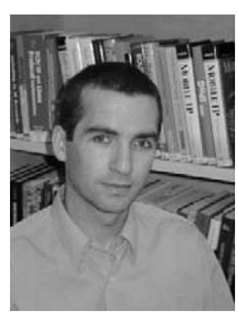
[19] J.L. Sobrinho, A.S. Krishnakumar, Real-time traffic over the IEEE 802.11 medium access control layer, Bell Labs Technical Journal 1 (2) (1996) 172–187.

[20] S. Chevrel, et al., Analysis and optimisation of the HIPERLAN Channel Access contention scheme, Wireless Personal Communications 4 (1) (1997) 27–39.

[21] IEEE 802.11e/D6.0, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Medium Access Control (MAC) enhancements for Quality of Service (QoS), Draft Supplement to IEEE 802.11 Standard, October 2003.

[22] A. Banchs, X. Pérez-Costa, D. Qiao, Providing Throughput Guarantees in IEEE 802.11e Wireless LANs, in: Proceedings of the 18th International Teletraffic Congress (ITC18), Berlin, Germany, September 2003.

[23] A. Banchs, L. Vollero, A delay model for IEEE 802.11e EDCA, IEEE Communications Letters 9 (6) (2005) 508–510.



Albert Banchs received his MSc and PhD degrees in Telecommunications from the Technical University of Catalonia in 1997 and 2002, respectively. His PhD received the national award for best thesis on Broadband Networks granted by the Professional Association of Telecommunication Engineers. He worked for the International Computer Science Institute, Berkeley, in 1997, for Telefonica I+D, Madrid, in 1998 and for NEC Network Laboratories, Heidelberg, from 1998 to 2003. Since 2003, he is with the University Carlos III of Madrid. His current research interests include resource allocation, QoS and performance evaluation of wireless and wired networks.

Pablo Serrano was born in Tarifa, Spain, on May 17, 1979. He received a MSc degree in Telecommunications from the University Carlos III of Madrid in 2002. Since that date he is a PhD candidate and a lecturer at the Telematics Department of the same university. His current research interests are performance evaluation and resource allocation of WLAN networks.

Arturo Azcorra received a MSc degree in Telecommunications from the Technical University of Madrid in 1986, and a PhD degree from the same university in 1989. His PhD received the national award for best thesis, jointly granted by the Professional Association of Telecommunication Engineers and the National Association of Electronic Industries. In 1993 he obtained an MBA from Instituto de Empresa, Madrid. He was a lecturer at the Technical University of Madrid from 1987 to 1990, when he was promoted to associate professor. In 1998 he joined the University Carlos III of Madrid, where he has stayed up to now as full professor. Since 1998 he is also deputy Vice-Provost for Academic Infrastructures at the University Carlos III of Madrid. Prof. Azcorra has been program committee member of several editions of IEEE-PROMS, IDMS, QofIS, CONEXT and IEEE-INFOCOM. He is general co-chair for CONEXT'05 and TPC co-chair for INFOCOM'06. His list of papers published in national and international journals, books and conferences is over 100 titles. Current research projects include broadband networks, multicast teleservices, active networks and advanced IP networks.