Optimal Configuration of 802.11e EDCA under Voice Traffic

Pablo Serrano, Albert Banchs and José Félix Kukielka
Universidad Carlos III de Madrid, Spain
E-mail: {pablo,banchs,kukielka}@it.uc3m.es

***Abstract*— In this paper we first present a model to analyze the average and standard deviation of the delay in a 802.11e EDCA WLAN under voice traffic. Next, based on this model, we compute the optimal configuration of the EDCA parameters. Specifically, with our optimal configuration we aim at admitting as many voice calls as possible while meeting a given quality criterion for the average delay and its typical deviation. Simulation results confirm the effectiveness of our optimal configuration; indeed, our configuration provides voice traffic with the desired quality criterion, and there exists no other configuration that could possibly admit more voice calls.**

I. Introduction

Wireless LANs (WLANs) have become a very popular technology for Internet access. The Medium Access Control algorithm used by today's WLANs is the one defined by the IEEE 802.11 standard. Recently, the IEEE 802 Working Group has approved a new standard called 802.11e [1] that extends the basic 802.11 algorithm with Quality of Service capabilities. This new standard is based on a number of open parameters whose configuration is yet an unsolved issue. Although the standard includes some recommendations for the parameters configuration, these recommendations are statically set and do no guarantee optimized performance.

In this paper we address the issue of finding the optimal configuration of a 802.11e EDCA WLAN under voice traffic. We first present a model of EDCA that, unlike previous analyses (see [2]–[4] and references therein), does not only account for the throughput and average delay characterization but also for the standard deviation of the delay. Indeed, we argue that variance is a fundamental measure in order to provide a real-time application such as voice traffic with meaningful QoS guarantees[1].

The second and main contribution of this paper is the proposal of a concrete algorithm for the configuration of the EDCA parameters for voice traffic. Our algorithm takes as input parameters the number of voice stations and the desired service quality criterion (namely, average delay and standard deviation), and provides as output the EDCA parameter values (if they exist) that satisfy this criterion. To our knowledge, this is the first attempt to compute analytically the optimal EDCA configuration for real-time traffic[2].

The rest of this article is structured as follows. Section II presents a brief description of the 802.11e EDCA Medium Access Control Algorithm. Section III analyzes the throughput, average delay and standard deviation of an EDCA WLAN under voice traffic; the analysis is validated by means of simulation results. Section IV is devoted to finding the optimal EDCA configuration; the effectiveness of the proposed configuration is validated also by means of simulation results. Finally, the article closes with some concluding remarks in Section V.

[1]In fact, given the average delay and its standard deviation, it is possible to provide guarantees on the delay distribution by means of the Tshebycheff inequality [5]. In this paper we do not further discuss this and simply assume that the average delay and the standard deviation are sufficient to provide voice traffic with the desired service guarantees.

[2]In [3] the optimal EDCA configuration for data traffic is computed.

II. 802.11e EDCA Medium Access Control Algorithm

The EDCA Medium Access Control algorithm is briefly summarized as follows. A station with a new packet to transmit monitors the channel activity. If the channel is idle for a period of time equal to the arbitration interframe space ($AIFS$) parameter, the station transmits. Otherwise, if the channel is sensed busy (either immediately or during the $AIFS$), the station continues to monitor the channel until it is measured idle for an $AIFS$, and, at this point, the backoff process starts.

Upon starting the backoff process, the station initializes its backoff time counter to a random value uniformly distributed in the range (0, $CW-1$), with CW initially set equal to the minimum contention window (CW_{min}) parameter. The backoff time counter is decremented once every slot time as long as the channel is sensed idle, "frozen" when a transmission is detected, and reactivated when the channel is sensed idle again for an $AIFS$ period.

Once the backoff counter reaches zero, the station accesses the channel. When it gains access to the channel, the station is allowed to transmit for a duration given by the transmission opportunity ($TXOP$) parameter. A collision occurs when two or more stations access the channel simultaneously. After a collision, CW is doubled (up to a maximum value equal to the CW_{max} parameter) and the backoff process is restarted. If the number of failed retries for a packet reaches a predetermined retry limit R, the packet is discarded. Once the backoff process is completed (either successfully or unsuccessfully), CW is set again to CW_{min}.

As it can be seen from the above description of EDCA, its operation depends on a number of configurable parameters (namely CW_{min}, CW_{max}, $AIFS$ and $TXOP$) which can be set to different values for different stations. The rest of this paper is devoted to finding the optimal configuration of these parameters for an EDCA WLAN under voice traffic.

III. Performance Analysis

In this section we analyze the performance of 802.11e EDCA under voice traffic as a first step towards finding the optimal EDCA configuration. We first present a number of considerations that allow us to fix the configuration of three of the parameters (CW_{max}, $AIFS$ and $TXOP$) and then we analyze the throughput, average delay and standard deviation of EDCA as a function of the remaining parameter (CW_{min}).

A. Considerations on the Configuration

The focus of this paper is on a WLAN operating under voice traffic. As in this scenario we only have one traffic class (namely voice), there is no need for introducing any type of differentiation, and the same EDCA parameter values can be assigned to all stations.

As a result of the above, we have that all the stations use the same $AIFS$ configuration. From this, it follows that the optimal setting for this parameter is its minimum possible value, namely $AIFS = DIFS$, as otherwise some extra time is unnecessarily lost after every transmission. This fixes the value of one of the four parameters.

We next consider the configuration of the CW_{max} parameter. When the number of stations in the channel is unknown, CW_{max} is typically set larger than CW_{min}, so that after a collision the CW increases and thus the probability of a new collision is reduced. However, this is not necessary in our case, as the number of stations is known and therefore their CW_{min} can be directly set so that the resulting collision probability corresponds to optimal operation. In addition, if we set CW_{max} larger than CW_{min}, the delay of the packets that suffer one or more collision drastically grows, which harms jitter performance. Based on these arguments, we set $CW_{min} = CW_{max}$, which fixes another parameter.

Given the stringent delay requirements of voice traffic, the parameters setting for voice stations will typically be chosen such that their transmission queue never grows to more than one packet (in particular, this holds for the configurations that we propose in Section IV). In the eventual case that queues grow above one packet, it is desirable that, upon accessing the channel, all waiting packets are transmitted in order to minimize their delay. To achieve this, in this paper we set the $TXOP$ parameter to its maximum allowed value. However, since according to the above reasoning this will occur rarely, hereafter we assume that (unless otherwise stated) stations only transmit one packet when they access the channel.

Based on the above considerations, we have that three out of the four parameters of EDCA are fixed (namely $AIFS$, CW_{max} and $TXOP$); the rest of the paper is devoted to finding the optimal configuration of the remaining parameter (CW_{min}).

B. Throughput Analysis

We next analyze the throughput performance of an EDCA WLAN with N voice stations as a function of the CW_{min} configuration. Following the behavior of many of today's most popular voice applications (like e.g. Skype[3]), which do not use silence suppression, we model voice stations as CBR traffic sources that generate a voice packet of size L every time interval T.

The key variable upon which we base the throughput analysis is τ, defined as the probability that a station transmits in a randomly chosen slot time. Based on this variable, the throughput r experienced by a given station is computed as follows:

$$r = \frac{P_g L}{P_s T_s + P_c T_c + P_e T_e} \quad (1)$$

where P_g is the probability that a randomly chosen slot time contains a successful transmission of the given station, P_s, P_c and P_e are the probabilities that a slot time contains a successful transmission, a collision or is empty, respectively, and T_s, T_c and T_e are the slot time durations in each case.

The above probabilities are computed as a function of τ as follows:

$$P_g = \tau(1-\tau)^{N-1} \quad (2)$$

$$P_s = N\tau(1-\tau)^{N-1} \quad (3)$$

$$P_e = (1-\tau)^N \quad (4)$$

$$P_c = 1 - P_e - P_s \quad (5)$$

Given $\tau \ll 1$, these probabilities can be accurately approximated by

$$P_g \cong \tau(1-(N-1)\tau) \quad (6)$$

$$P_s \cong N\tau(1-(N-1)\tau) \quad (7)$$

$$P_e \cong (1-N\tau) \quad (8)$$

$$P_c \cong 1 - N\tau(1-(N-1)\tau) - (1-N\tau) \quad (9)$$

From the above, we have a formula to compute the throughput as a function of τ, $r(\tau)$. Based on this, we can obtain the throughput performance as a function of CW_{min} as follows. We say that a station is saturated when it always has packets ready for transmission. The τ value of such a station will be [3]

$$\tau_{sat} = \frac{2}{CW_{min}+1} \quad (10)$$

If, for a given CW_{min} configuration we have that $r(\tau_{sat}) < L/T$, then stations will be saturated[4], as their incoming rate L/T will be larger than the outgoing rate $r(\tau_{sat})$. Throughput performance in this case will be the given by $r(\tau_{sat})$.

On the other hand, if the CW_{min} configuration is such that $r(\tau_{sat}) \geq L/T$, then stations will not be saturated and their throughput will be equal to the incoming rate, L/T.

C. Analysis of the τ of Operation

Based on the above throughput analysis, we now analyze the τ value at which stations operate.

In case of saturation, the value of τ is directly given by Eq. (10). In case of non saturation, the throughput experienced by the stations is equal to their incoming rate, and therefore the τ of operation has to satisfy the following second order equation:

$$r(\tau) = L/T \quad (11)$$

[3]See http://www1.cs.columbia.edu/~salman/publications/skype1_4.pdf

[4]The reader is referred to [4] for a more detailed discussion on the throughput behavior as a function of the CW_{min} configuration.

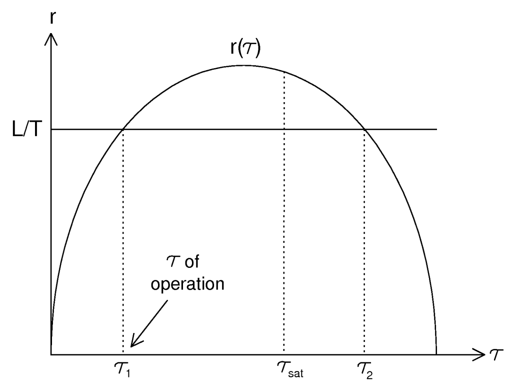

Fig. 1. τ of operation.

From Figure 1, which plots $r(\tau)$ as a function of τ, it can be seen that the above equation has two solutions: τ_1 and τ_2. We next show that the τ of operation corresponds to the smallest of the two, i.e. τ_1.

From the fact that under non saturation $r(\tau_{sat}) > L/T$, we have that the value of τ_{sat} surely falls between τ_1 and τ_2. Note that τ_{sat} corresponds to the extreme case when a station always has packets ready for transmission and only waits one backoff process between each transmission and the next one. Therefore, τ_{sat} represents an upper bound on the maximum τ at which the station can possibly operate. As a consequence of this reasoning, we have that τ_2 cannot be the point of operation, which leaves τ_1 as the only possible solution.

D. Average Delay Analysis

We next analyze, as a function of the τ of operation obtained in the previous section, the delay performance of the WLAN. Specifically, our focus is on the time elapsed between the beginning of the backoff process and the successful transmission of a packet. Given our assumption of Section III-A that EDCA parameters are set such that transmission queues do not grow to more than one packet, this corresponds to the total delay of the WLAN.

We start by analyzing the average value of the delay. This can be computed as follows:

$$E[d] = \sum_{j=0}^{R} P_{tx}(j)E[d_j] \tag{12}$$

where $P_{tx}(j)$ is the probability that a packet is successfully transmitted after j retries and $E[d_j]$ is the expected delay in this case. $P_{tx}(j)$ is computed as

$$P_{tx}(j) = (1-p)p^j \tag{13}$$

where p is the probability that a transmission attempt collides, which is given by

$$p = 1-(1-\tau)^{N-1} \tag{14}$$

$E[d_j]$ is computed as follows:

$$E[d_j] = T_s + jT_c + E[d_{bo}(j)] \tag{15}$$

where $E[d_{bo}(j)]$ is the total time spent in average with backoff counter decrements for the case of j collisions. This is calculated as

$$E[d_{bo}(j)] = jE[c_{bo}]E[T_{slot}] \tag{16}$$

where $E[c_{bo}]$ is the expected backoff time counter drawn at the beginning of a backoff process and $E[T_{slot}]$ is the average duration of a slot time when the considered station does not transmit.

Since the backoff time counter is calculated from a uniform distribution between 0 and $CW_{min}-1$, $E[c_{bo}]$ is equal to

$$E[c_{bo}] = \frac{CW_{min}-1}{2} \tag{17}$$

Finally, $E[T_{slot}]$ is calculated as follows by noting that during those slot times the considered station does not transmit:

$$E[T_{slot}] = P_{e,N-1}T_e + P_{s,N-1}T_s + P_{c,N-1}T_c \tag{18}$$

where

$$P_{e,N-1} = (1-\tau)^{N-1} \tag{19}$$

$$P_{s,N-1} = (N-1)\tau(1-\tau)^{N-2} \tag{20}$$

$$P_{c,N-1} = 1 - P_{e,N-1} - P_{s,N-1} \tag{21}$$

which terminates the analysis of the average delay.

E. Delay Standard Deviation Analysis

Next, we analyze the standard deviation of the delay. The analysis follows the same lines as the computation of the average delay in the previous section.

The standard deviation of the delay can be computed as a function of the first and second moments of the delay as follows:

$$\sigma_d = \sqrt{E[d^2] - E[d]^2} \tag{22}$$

$E[d]$ has already been computed above. To compute $E[d^2]$, we proceed similarly as in Eq. (12):

$$E[d^2] = \sum_{j=0}^{R} P_{tx}(j)E[d_j^2] \tag{23}$$

$P_{tx}(j)$ has already been obtained in Eq. (13). By definition, $E[d_j^2]$ can be expressed as

$$E[d_j^2] = E[d_j]^2 + \sigma_{d_j}^2 \tag{24}$$

where $E[d_j]$ has already been computed in Eq. (15).

The remaining challenge is the computation of $\sigma_{d_j}^2$. Since T_s and T_c are constants, from Eq. (15) it follows

$$\sigma_{d_j}^2 = \sigma_{d_{bo}(j)}^2 \tag{25}$$

Since in case of j retransmission, the total backoff delay is composed of j backoff components, we have

$$\sigma_{d_{bo}(j)}^2 = j\sigma_{d_{bo}}^2 \tag{26}$$

where $\sigma_{d_{bo}}$ can be expressed as

$$\sigma_{d_{bo}}^2 = E[d_{bo}^2] - E[d_{bo}]^2 \tag{27}$$

$E[d_{bo}]$ has already been obtained above. $E[d_{bo}^2]$ can be calculated as

$$E[d_{bo}^2] = \sum_{k=0}^{CW_{min}-1} P_{bo}(k) E[(\underbrace{T_{slot} + T_{slot} + \cdots + T_{slot}}_{k \text{ times}})^2] \quad (28)$$

where $P_{bo}(k) = 1/CW_{min}$ is the probability that the backoff counter drawn is equal to k and

$$E[(\underbrace{T_{slot} + T_{slot} + \cdots + T_{slot}}_{k \text{ times}})^2] = k^2 E[T_{slot}]^2 + k\sigma_{T_{slot}}^2 \quad (29)$$

Finally, by combining the above two equations,

$$\begin{aligned} E[d_{bo}^2] &= \frac{E[T_{slot}]^2}{CW_{min}} \sum_k k^2 + \frac{\sigma_{T_{slot}}^2}{CW_{min}} \sum_k k \\ &= E[T_{slot}]^2 \frac{(CW_{min}-1)(2CW_{min}-1)}{6} \\ &+ \sigma_{T_{slot}}^2 \frac{CW_{min}-1}{2} \end{aligned} \quad (30)$$

where

$$\sigma_{T_{slot}}^2 = E[T_{slot}^2] - E[T_{slot}]^2 \quad (31)$$

$$E[T_{slot}^2] = P_{e,N-1}T_e^2 + P_{s,N-1}T_s^2 + P_{c,N-1}T_c^2 \quad (32)$$

which terminates the delay standard deviation analysis.

F. Validation

We validated the accuracy of our analysis by comparing analytical results against simulations. For the simulations, we used an event-driven simulator that closely follows the 802.11e EDCA behavior for each station. The experiments were performed for a WLAN with the system parameters of the IEEE 802.11b physical layer. Following the behavior of standard PCM codecs, voice sources generated one 80 byte packet every 10 ms.

Figures 2 and 3 plot the average and standard deviation of the backoff delay for different configurations of the CW_{min} parameter as well as different numbers of voice stations. The three values chosen for the number of voice stations, $N \in \{10, 15, 20\}$, correspond to a low, medium and heavy loaded WLAN, respectively. Simulation results are plotted with 95% confidence intervals, although these are so small that can barely be appreciated in the graphs.

From the figures, we observe that analytical results match simulations remarkably well, which confirms the accuracy of our analysis. We further observe that delays show the following behavior:

- For too low CW_{min} values, the WLAN is saturated and delays are very large.
- As CW_{min} increases, after crossing a certain threshold (which varies for different N values) the WLAN leaves saturation and delays decrease sharply.
- After this threshold, delays increase gradually with the CW_{min}. The reason for this gradual increase is that, the larger the CW_{min}, the longer the completion of the backoff process takes.

From the above, it can be intuitively seen that the CW_{min} values that provide the best performance are the ones close to the saturation threshold. In the following, we address the issue of finding this optimal configuration.

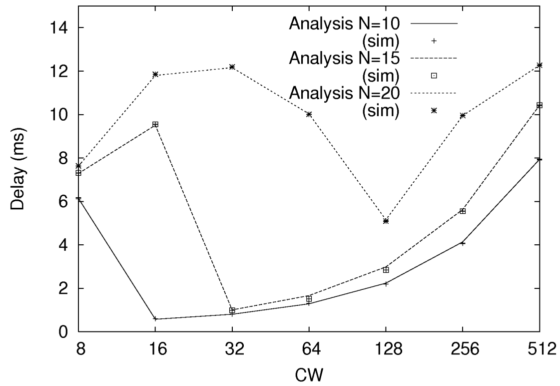

Fig. 2. Validation of the average delay.

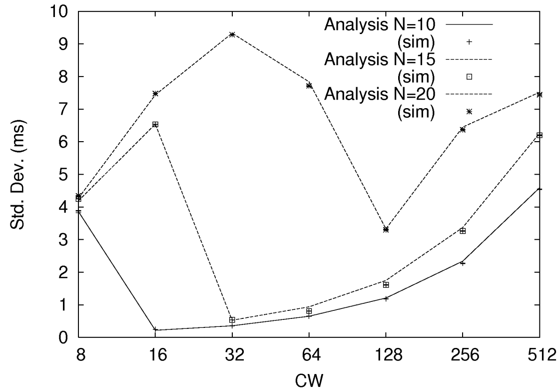

Fig. 3. Validation of the delay standard deviation.

IV. Optimal Configuration

In this section we present an algorithm that, given the desired performance for voice traffic, finds the optimal configuration that satisfies this quality criterion. Specifically, our algorithm takes as input the desired upper bound values for the average delay and its standard deviation (D_{max} and σ_{max}) and provides the following output: i) it determines if there exists any CW_{min} configuration that meets the given requirements, and ii) if it exists, it gives the optimal CW_{min} configuration.

In the following, we first obtain some lower and upper bounds for CW_{min} and then, based on these bounds, we propose an algorithm to calculate the optimal CW_{min}.

A. Bounds for CW_{min}

We start by analyzing the CW_{min} range that provides good throughput performance. According to Section III-B, the WLAN will not be saturated as long as CW_{min} is set such that the following condition holds: $r(\tau_{sat}) \geq L/T$, where τ_{sat} is a function of CW_{min} as given by Eq. (10).

For any CW_{min} that does not meet the above condition, the outgoing rate will be smaller than the incoming one and

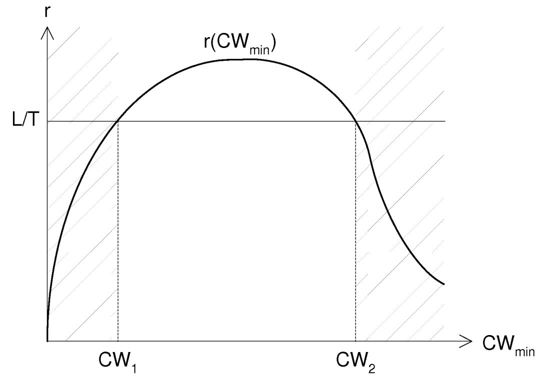

Fig. 4. CW_{min} bounds for throughput.

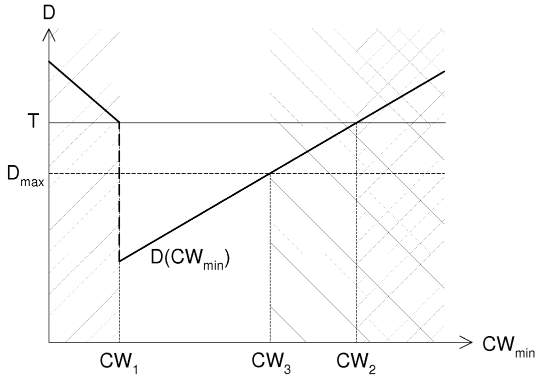

Fig. 5. CW_{min} bound for average delay.

as a result throughput performance will be degraded. As it can be observed from Figure 4, this imposes a lower and an upper bound on CW_{min}. Hereafter, we refer to these bounds as CW_1 and CW_2, respectively.

We now analyze the CW_{min} range to meet the given delay performance requirements. According to the average delay analysis of Section III-D, as long as the WLAN is not saturated (which is given by the above bounds) average delay is an increasing function of CW_{min}. As a result, the requirement that average delay cannot exceed a given D_{max} value imposes an additional upper limit on CW_{min}, which we refer to with CW_3. Indeed, as it can be seen from Figure 5, for any CW_{min} value larger than CW_3 the average delay will not meet the given criterion.

Following a similar reasoning as above, we have that the requirement on the delay standard deviation imposes yet an additional upper limit, which we refer to with CW_4.

B. Optimal CW_{min} Configuration

We next propose an algorithm to compute the optimal CW_{min} based on the lower bound (CW_1) and three upper bounds (CW_2, CW_3 and CW_4) obtained above.

From the previous section, we have that any CW_{min} that falls within the bounds meets the given quality criterion. The remaining challenge is to choose one CW_{min} value within this range. Based on the following argument, we choose the largest possible value. As it can be observed from Figure 5, in the given range delay performance improves as CW_{min} decreases. The problem, however, is that as CW_{min} approaches CW_1, there is the risk of suffering a sharp performance decrease. In order to avoid this, we choose the CW_{min} value that, while meeting the given criterion, falls as far as possible from this critical point.

We next present our algorithm resulting from all the above considerations. Note that the algorithm is extremely efficient as each of the steps only involves the calculation of one equation of first or second order:

- In the first step, we compute CW_1 and CW_2 by solving $r(CW_{min}) = L/T$, using the expression for $r(CW_{min})$ obtained in Section III-B with the τ of operation given in Section III-C for saturation.
- Next, we compute CW_3 by solving $E[d] = D_{max}$, using the $E[d]$ expression of Section III-D with the τ of operation of Section III-C for non saturation.
- We then obtain CW_4 by solving $\sigma_d = \sigma_{max}$, using in this case the expression for σ_d of Section III-E.
- As a final step, the algorithm compares the lower bound (CW_1) with the minimum of all upper bounds (CW_2, CW_3 and CW_4): if $CW_1 > min(CW_2, CW_3, CW_4)$, there exists no CW_{min} value that satisfies the desired quality criterion and the algorithm indicates that it is not possible to admit the given number of voice calls.
- Otherwise, the algorithm terminates by giving the following optimal configuration: $CW_{min} = min(CW_2, CW_3, CW_4)$.

C. Validation

We validated our algorithm by comparing the performance of our configuration given by the algorithm ($CW_{algorithm}$) against the result of performing an exhaustive search over the CW_{min} space ($CW_{exhaustive}$). Specifically, for the exhaustive search we evaluated by means of simulation the delay performance of all possible CW_{min} values and took the largest CW_{min} that met the given quality criterion. We performed this experiment for three different quality criteria ranging from a more stringent criterion ($D_{max} = \sigma_{max} = 2.5ms$) to a more relaxed one ($D_{max} = \sigma_{max} = 5ms$). For the simulations, the $TXOP$ parameter was taken equal to its maximum value.

Simulation results are presented in Table I. It can be seen that the proposed configuration is always very close to the one obtained from the exhaustive search. In all three experiments, our algorithm admits as many voice calls as the exhaustive search (20 for the first two experiments and 19 for the third one). In addition, the desired quality criteria are always met by our configuration. We conclude that our algorithm is effective in admitting as many voice calls as possible while guaranteeing the desired performance.

V. Conclusions

In this paper we have proposed an algorithm to compute the optimal configuration of EDCA for voice traffic. To our knowledge, this is the first attempt to compute analytically the configuration of EDCA for delay sensitive traffic. The optimal configuration values that we have obtained are about one order

TABLE I

ALGORITHM VALIDATION.

D_{max}	σ_{max}	N	$CW_{algorithm}$	$D_{algorithm}$	$\sigma_{algorithm}$	$CW_{exhaustive}$	$D_{exhaustive}$	$\sigma_{exhaustive}$
5 ms	5 ms	10	314	4.95	2.78	317	4.99	2.82
		15	225	4.91	2.87	229	4.99	2.92
		20	118	4.72	3.02	125	4.99	3.25
5 ms	2.5 ms	10	274	4.35	2.43	281	4.45	2.49
		15	186	4.07	2.36	196	4.28	2.49
		20	89	3.65	2.48	91	4.31	2.49
2.5 ms	2.5 ms	10	145	2.45	1.32	148	2.49	1.35
		15	104	2.32	1.29	111	2.47	1.39
		19	66	2.29	1.42	72	2.49	1.54

of magnitude above the 802.11e standard recommendation [1] ($CW_{min} = 8$), which poses doubts on the optimality of this recommendation.

REFERENCES

[1] IEEE 802.11e, *MAC Enhancements for QoS*. Supplement to IEEE 802.11 Standard, November 2005.

[2] G.-R. C. *et al.*, "Performance Analysis of Finite Load Sources in 802.11b Multirate Environments," *Computer Communications*, June 2005.

[3] A. Banchs and L. Vollero, "Throughput Analysis and Optimal Configuration of EDCA," *Computer Networks*, August 2006.

[4] P. S. *et al.*, "Performance Anomalies of nonoptimally configured WLANs," in *Proceedings of WCNC*, April 2006.

[5] P. et al., *Probability, Random Variables and Stochastic Processes*. McGraw-Hill, December 2001.