

# Providing Throughput Guarantees in IEEE 802.11e Wireless LANs

Albert Banchs<sup>a</sup>, Xavier Pérez-Costa<sup>a</sup>, Daji Qiao<sup>b</sup>

<sup>a</sup>Network Laboratories, NEC Europe Ltd., Heidelberg, Germany

<sup>b</sup>Real-Time Computing Laboratory, The University of Michigan, Ann Arbor, MI, USA

The upcoming new standard IEEE 802.11e aims at providing Quality of Service (QoS) support in 802.11 Wireless LANs. While the QoS mechanisms in 802.11e, namely the EDCF and the HCF, have already been defined in the standard draft, the challenge lies in the configuration of these mechanisms in order to provide the desired services. In this paper, we deal with the configuration of the EDCF in order to provide throughput guarantee services. We derive an analytical model for the throughput performance of an 802.11e Wireless LAN under the EDCF, and, based on this analysis, we propose an admission control and parameter configuration algorithm that (1) provides the committed throughput guarantees; and (2) accepts as many requests as possible. The performance of the proposed algorithm is evaluated via simulation.

## 1. INTRODUCTION

The IEEE 802.11 Wireless LAN standard [1] has been accepted widely in many different environments today [2]. In its current version, the 802.11 standard can be considered as an extension to Ethernet, which supports only Best-Effort services. Recently, the interest in wireless networks supporting Quality-of-Service (QoS) has grown tremendously. Accordingly, the IEEE 802.11 Working Group established an activity to enhance the current 802.11 MAC protocol to support applications with QoS requirements.

The discussion of the schemes for QoS support in 802.11 is in its way of being completed by the IEEE 802.11 Task Group E (TGe). The work presented in this paper is based on the 802.11e supplement standard as specified in the draft by TGe [3]. 802.11e introduces two medium access control mechanisms: the Enhanced Distributed Coordination Function (EDCF) and the Hybrid Coordination Function (HCF). The challenge with 802.11e lies in determining how to configure the EDCF and the HCF to provide the desired services. Providing guaranteed throughputs is widely accepted as one of the desirable services in a QoS architecture [4]. This service suits well, e.g., the needs of data communications. In line with these considerations, in this paper we investigate how to provide throughput guarantee services in an 802.11e Wireless LAN under the EDCF.

The rest of this paper is organized as follows. In Section 2, we introduce 802.11e EDCF and describe our approach to provide throughput guarantees with EDCF. Our admission control and parameter configuration algorithm is presented in Section 3. The algorithm is validated via simulation in Section 4. Finally, Section 5 concludes the paper.

## 2. SYSTEM OVERVIEW

This section gives an overview of the solution that we propose to provide throughput guarantees in Wireless LANs. Our approach is based on properly configuring the open parameters of the 802.11e EDCF. We first give a brief introduction to the EDCF. Then, we present an architecture to configure the EDCF parameters. Finally, we discuss which open parameters of the EDCF are more suitable for providing throughput guarantee services.

### 2.1. The IEEE 802.11e EDCF

In the 802.11e EDCF, QoS support is realized with the introduction of Backoff Instances (BIs). Packets can be delivered through multiple BIs within one station, and each BI is configured with its own parameters. Each BI contends for transmission and starts a backoff independently after detecting the channel being idle for an Arbitration Inter-Frame Space (AIFS), where the AIFS is a parameter dependent on the BI. After waiting for AIFS time, the BI sets the backoff counter to a random number drawn from the interval  $[0, CW]$ , where  $CW$  is the contention window size.  $CW$  is initially set to  $CW_{min}$ , the minimum contention window size, which is another parameter dependent on the BI. The backoff counter is decremented once every slot time. When the medium is determined busy before the counter reaches zero, the BI has to wait for the medium being idle for AIFS time again before continuing to count down the counter. When the counter finally reaches zero, the BI transmits its packet.

A collision occurs when two or more stations start transmission simultaneously in the same time slot. An acknowledgment is sent by the receiving station after it receives a packet successfully and waits for SIFS (Short Inter-Frame Space) time. Only after receiving an acknowledgment correctly, the transmitting station assumes successful delivery of the transmitted packet. In the case of an unsuccessful packet transmission, a new contention window value  $CW_{new}$  is calculated according to  $CW_{new} = (CW_{old} + 1) \cdot PF - 1$ , where the persistence factor  $PF$  is yet another parameter of the BI and  $CW_{old}$  is the previous value of the contention window. The backoff counter is then set to a random number drawn from the interval  $[0, CW_{new}]$  and the backoff process is re-entered.  $CW_{new}$  is larger than  $CW_{old}$  in order to reduce the probability of a new collision. Besides, the  $CW$  value never exceeds  $CW_{max}$ , the maximum contention window size, which is another parameter of the BI.

The use of the Request-To-Send (RTS)/Clear-To-Send (CTS) mechanism is optional in 802.11e. When this option is applied, upon the backoff counter reaching zero, the transmitting station sends an RTS packet instead of a data packet to the receiving station, which responds with a CTS packet. The data packet is sent when the transmitting station receives the CTS packet. RTS/CTS is used to handle the “hidden nodes” scenario and it also increases the bandwidth efficiency because, if collisions occur, they do not occur to the long data packets but to relative shorter control packets. In this paper, we assume that the RTS/CTS option is turned off. However, the work presented here could be easily extended to the RTS/CTS case.

### 2.2. Throughput Guarantee Service and Architecture

Our goal with this paper is to come up with a configuration of the 802.11e EDCF to provide a BI with a guaranteed throughput. Without loss of generality, in the rest of this

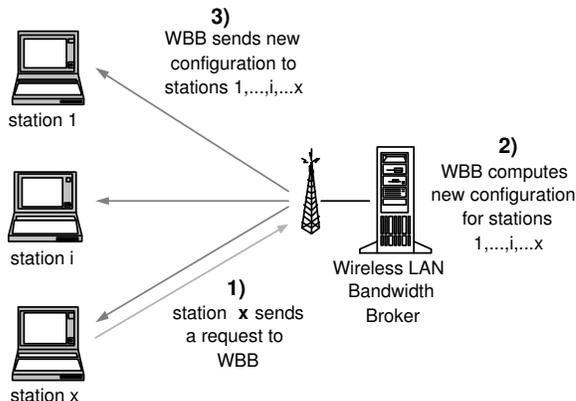


Figure 1. Throughput Guarantee Service Architecture

paper we will assume that there exists only one BI for each station, and we will refer to the throughput guaranteed to the BI as the station’s guaranteed throughput.

Our algorithm for providing throughput guarantees is based on a centralized control entity that we call the Wireless LAN Bandwidth Broker (WBB). The WBB functionality could be located, e.g., at the Access Point (AP). When an 802.11e station has a throughput requirement, it sends the corresponding request to the WBB, which computes whether the request can be granted. In the affirmative case, the WBB computes the new configuration for the accepted stations that leads to the committed throughput guarantees, and then distributes the computed configuration parameters among the stations. In the negative case, the request is rejected and the configuration of the accepted stations remains unchanged. In both cases, only the accepted stations are allowed to send data packets, i.e., we do not consider Best-Effort services in this paper; in the following section we outline our ideas for extending the architecture presented in this paper to support Best-Effort services. Fig. 1 illustrates our architecture for providing throughput guarantee services.

### 2.3. Discussion on the EDCF Open Parameters

The challenge in order to implement the architecture described above remains on how to optimally compute the various configuration parameters of the accepted stations in order to (1) provide the requested throughput guarantees; and (2) accept as many requests as possible.

In the description of the EDCF given in Section 2.1, we have seen that the service received by a station depends on the following four parameters assigned to the station: the Arbitration Inter-Frame Space (AIFS), the minimum contention window size ( $CW_{min}$ ), the maximum contention window size ( $CW_{max}$ ), and the Persistence Factor ( $PF$ ). These four parameters offer a very large degree of freedom, which makes it very difficult to find an optimum solution. Therefore, it is necessary to reduce this freedom based on additional considerations.

In the EDCF, after each unsuccessful packet transmission, the contention window is increased in order to reduce the probability of a new collision. However, this is not nec-

essary in our architecture, since the number of active stations is known to the WBB and their contention windows can be computed such that the resulting collision probability leads to an optimum channel utilization. Therefore, we set  $PF = 1$  resulting in  $CW_{min} = CW_{max} = CW$ , which leaves us only two open parameters:  $CW$  and AIFS.

As indicated in [5], different  $CW$  values result in finely-tuned levels of service differentiation, while different AIFS provide much stronger differentiation. Based on this observation, we have chosen to keep AIFS constant, i.e.,  $AIFS = DIFS$ , and provide the throughput guarantee services, which require a fine tuning in the resulting throughput distribution, by means of different  $CW$  values. Using different AIFS would be more appropriate for supporting Best-Effort services in addition to throughput guarantee services, since the coexistence of these two types of services should be based on giving the highest possible priority to the latter. Specifically, we envisage supporting Best-Effort services in our architecture by assigning large AIFS and  $CW$  values to the Best-Effort stations; with such a configuration, Best-Effort services only consume the leftover bandwidth and have a negligible impact onto our throughput guarantee services. A more detailed analysis of Best-Effort services support within our architecture has been left as part of the future work.

### 3. THROUGHPUT ANALYSIS AND ADMISSION CONTROL

The algorithm for deciding whether a new request can be accepted and for computing the  $CW$  values of the accepted stations has been left open in the previous section. In this section, we analyze the throughput performance of an IEEE 802.11e Wireless LAN under the EDCF, and present our admission control and  $CW$  computation algorithm based on this analysis.

#### 3.1. Source and Channel Models

We assume that all stations are greedy, i.e., always have packets to transmit. We argue that this source model is appropriate to study throughput guarantees because (1) a target station that is not able to send all the packets it generates always has packets to transmit, and (2) having all other stations with always packets to transmit (hence, always contending for the shared wireless medium) represents the worst case scenario for the throughput of the target station.

Therefore, a station is guaranteed either (1) to be able to send all the packets it generates, in which case throughput guarantees are not necessary, or (2) to experience a throughput larger than or equal to the calculated with our source model; hence, our source model results in the guaranteed throughput.

Furthermore, we assume no channel errors and all stations within a communication distance of each other (i.e. no hidden nodes).

#### 3.2. Throughput Analysis

Consider the scenario when there are  $n$  greedy contending stations — as defined above — in the Wireless LAN. Assume that station  $i$  uses a contention window  $CW_i$  to access the wireless medium. As given in [6], the probability that station  $i$  transmits in a generic

time slot is

$$\tau_i = \frac{2}{CW_i + 1}, \quad (1)$$

and the probability of a successful packet transmission from station  $i$  is

$$P_{s,i} = \tau_i \cdot \prod_{j \neq i} (1 - \tau_j). \quad (2)$$

Define  $\omega_i$  as

$$\omega_i = \frac{\tau_i}{\tau_1}, \quad (3)$$

where we are taking station 1 as reference.

The probability of a successful packet transmission,  $P_s$ , is given by

$$P_s = \sum_i \tau_i \prod_{j \neq i} (1 - \tau_j) = \sum_i \omega_i \tau_1 \prod_{j \neq i} (1 - \omega_j \tau_1), \quad (4)$$

and the probability of an empty time slot,  $P_e$ , is given by

$$P_e = \prod_i (1 - \tau_i) = \prod_i (1 - \omega_i \tau_1). \quad (5)$$

The aggregate throughput can then be calculated as [7]

$$r_{total} = \frac{P_s \cdot \ell}{P_s \cdot T_s + P_e \cdot T_e + P_c \cdot T_c}, \quad (6)$$

where  $\ell$  is the average payload length,  $T_s$  is the average duration of a successful transmission,  $T_c$  is the average duration of a collision,  $T_e$  is the duration of an idle time slot (*tSlotTime*), and  $P_c$  is the collision probability, which is given by

$$P_c = 1 - P_s - P_e. \quad (7)$$

Eq. (6) can be rewritten as

$$r_{total} = \frac{\ell}{T_s - T_c + \frac{P_e \cdot (T_e - T_c) + T_c}{P_s}}. \quad (8)$$

The ratio between the throughputs of two stations  $i$  and  $j$  is equal to

$$\frac{r_i}{r_j} = \frac{\tau_i \cdot \prod_{k \neq i} (1 - \tau_k)}{\tau_j \cdot \prod_{k \neq j} (1 - \tau_k)} = \frac{\tau_i \cdot (1 - \tau_j)}{\tau_j \cdot (1 - \tau_i)}. \quad (9)$$

Since  $\tau_i$  is normally a small value, the above equation is approximately equivalent to

$$\frac{r_i}{r_j} = \frac{\tau_i}{\tau_j}, \quad (10)$$

and this approximation becomes more accurate as  $n$  gets larger, which is usually our region of interest. Specifically, we would like our approximation to be accurate when the number of the accepted stations almost saturate the channel, since this is the region

where we may actually have to start rejecting stations. This situation will typically occur when the number of the accepted stations is already high, which is precisely when our approximation is more accurate.

From the above it follows

$$\frac{r_i}{r_j} = \frac{\omega_i}{\omega_j} \implies r_i = \frac{\omega_i}{\sum_i \omega_i} \cdot r_{total}. \quad (11)$$

Finally, substituting Eq. (8) into the above leads to the following expression for the throughput experienced by station  $i$ :

$$r_i = \frac{\omega_i}{\sum_i \omega_i} \cdot \frac{\ell}{T_s - T_c + \frac{P_e \cdot (T_e - T_c) + T_c}{P_s}}, \quad (12)$$

which concludes the throughput analysis.

### 3.3. The Throughput Requirements Problem

Based on the above throughput analysis, we now deal with the problem of finding, if it exists, a contention window set  $\{CW_1, \dots, CW_n\}$  that meets the throughput requirements for the  $n$  stations, i.e.,  $\forall i \in \{1, \dots, n\}$ ,  $r_i \geq R_i$ , where  $r_i$  is the throughput actually experienced by station  $i$  and  $R_i$  is the throughput requirement of the station.

We say that a throughput requirement set  $\{R_1, \dots, R_n\}$  is admissible if we can find a contention window set that satisfies the requirements. The region formed by the admissible points  $\{R_1, \dots, R_n\}$ 's is called the admissibility region. According to our goals stated in Section 2, our admission control algorithm should accept as many requests as possible, i.e., it should maximize the admissibility region. By finding a solution to the throughput requirements problem whenever a solution exists, we maximize the admissibility region.

We can restrict our search to the solutions that comply

$$\frac{r_i}{r_j} = \frac{R_i}{R_j}, \quad (13)$$

since it can be proven that, if there exists a solution to the throughput requirements problem that does not comply the above restriction, then a solution complying the restriction also exists. Specifically, let  $\{CW_1, \dots, CW_n\}$  be a set of contention windows that solves the throughput requirements problem but does not comply Eq. (13). Then, it can be seen that the new set  $\{CW'_1, \dots, CW'_n\}$ , where  $CW'_j$  is equal to  $CW_j$ , being  $j$  the station for which  $r_j/R_j$  is minimum, and any other  $CW'_i$  is such that Eq. (13) is complied, also solves the throughput requirements problem.

With the above, the throughput requirements problem can be reformulated as to find the contention window set that maximizes simultaneously all  $r_i$ 's subject to the condition of Eq. (13) — if this set does not solve the throughput requirements problem, then any other set complying Eq. (13) will result in smaller  $r_i$ 's and therefore will not solve the problem either. With the approximation of Eq. (10), the constraint of Eq. (13) is equivalent to

$$\omega_i = \frac{R_i}{R_1}, \quad (14)$$

which fixes the  $\omega_i$ 's of all stations.

Recall Eq. (12):

$$r_i = \frac{\omega_i}{\sum_i \omega_i} \cdot \frac{\ell}{T_s - T_c + \frac{P_e \cdot (T_e - T_c) + T_c}{P_s}}. \quad (15)$$

Since  $\ell$ ,  $T_s$ , and  $T_c$  are constants, maximizing the following expression will result in the maximization of all  $r_i$ 's:

$$\hat{r} = \frac{P_s}{P_e \cdot (T_e - T_c) + T_c} = \frac{\sum_i \omega_i \tau_1 \prod_{j \neq i} (1 - \omega_j \tau_1)}{\prod_i (1 - \omega_i \tau_1) (T_e - T_c) + T_c}. \quad (16)$$

Under the assumption of  $\tau_1 \ll 1$  — which is true when  $n$  is large enough — we can make the following approximation:

$$\hat{r} \approx \frac{a\tau_1 - b\tau_1^2}{c\tau_1 + T_e}, \quad (17)$$

where

$$a = \sum_i \omega_i, \quad b = \sum_i \sum_{j \neq i} \omega_i \omega_j \quad \text{and} \quad c = \sum_i \omega_i \cdot (T_c - T_e). \quad (18)$$

The optimal value of  $\tau_1$ ,  $\tau_1^*$ , that maximizes  $\hat{r}$  can then be obtained by

$$\left. \frac{d \hat{r}}{d \tau_1} \right|_{\tau_1 = \tau_1^*} = 0 \implies bc \cdot (\tau_1^*)^2 + 2bT_e \cdot \tau_1^* - aT_e = 0 \implies \tau_1^* = \frac{\sqrt{(bT_e)^2 + abcT_e} - bT_e}{bc}. \quad (19)$$

Consequently, the optimal contention window set that maximizes simultaneously all  $r_i$ 's is

$$\forall i, \quad CW_i^* = \frac{2}{\tau_i^*} - 1 = \frac{2}{\omega_i \tau_1^*} - 1, \quad (20)$$

which closes the throughput requirements problem.

### 3.4. Admission Control and CW Computation Algorithm

Assume that an IEEE 802.11e Wireless LAN is operating with a contention window set  $\{CW_1, \dots, CW_n\}$  that meets the throughput requirements for  $n$  stations, i.e.,  $\forall i \in \{1, \dots, n\}$ ,  $r_i \geq R_i$ , where  $r_i$  is the throughput actually experienced by station  $i$ .

When a new station ( $n+1$ ) with throughput requirement  $R_{n+1}$  would like to join the network, the WBB first computes a new contention window set  $\{CW'_1, \dots, CW'_n, CW'_{n+1}\}$  using Eq. (20). Then, it uses Eq. (12) to compute the throughputs that the  $n+1$  stations would receive with this new contention window set. If the resulting throughputs meet the requirements, i.e.,

$$\forall i \in \{1, \dots, n+1\}, \quad r'_i \geq R_i, \quad (21)$$

then station ( $n+1$ ) is accepted to the network and the new contention window set is distributed to all the stations. Otherwise, station ( $n+1$ ) is rejected, because, according to the analysis provided in Section 3.3, if the contention window set  $\{CW'_1, \dots, CW'_n, CW'_{n+1}\}$  does not meet the throughput requirements, then there exists no contention window set that could possibly meet the requirements.

## 4. SIMULATION RESULTS AND DISCUSSIONS

In this section, we evaluate the effectiveness of our proposed admission control and parameter configuration algorithm via simulation. We simulated an 802.11e Wireless LAN that consists of a number of wireless stations communicating with the WBB. The packet length was set to 1000 bytes for all simulations and the packets were transmitted at 2Mbps. The simulations were performed in the ns-2 simulator [8].

Following the same rationale as discussed in Section 3.1, the traffic model used for all simulations has been UDP CBR with a sending rate high enough such that all stations always have packets to transmit. Note that using different traffic models (e.g., TCP or UDP ON/OFF) would have an impact onto the resulting throughput. The study of such impact has been left as part of the future work; however, based on our previous results from [9], we expect this impact to be small.

### 4.1. Committed Throughput Guarantees

In order to assess that our approach meets the committed throughput guarantees, we simulated the scenario corresponding to  $n$  stations in the Wireless LAN with homogeneous throughput requirements ( $R$  for all stations) and using the  $CW$  values given by our algorithm. Fig. 2 shows the throughput obtained by each station in this scenario as a function of  $n$ , according to both the analysis and the simulation. We can see that, for the throughput guarantee of  $R = 100\text{Kbps}$ , the maximum number of stations that our admission control algorithm can accept is  $n_{max} = 16$ . Simulation results confirm that the throughput received by each station,  $\text{simu\_TH}$ , for this value of  $n$  is greater than 100Kbps (see also in Table 1). We obtain similar results for  $R = 200\text{Kbps}$ ; in this case, the maximum number of stations that can be accepted is  $n_{max} = 8$ . Hence, we conclude that our admission control and parameter configuration algorithm is effective in providing the committed throughput guarantees to the accepted stations.

### 4.2. Admissibility Region

In the above scenario we have seen that, for a given throughput request  $R$ , our algorithm can accept up to  $n_{max}$  stations. In order to verify our goal of maximizing the admissibility region, we ran a set of simulations with  $(n_{max}+1)$  stations in the network and all possible  $CW$  values for a given throughput requirement  $R$ . If the largest throughput obtained from this set of simulations is above  $R$ , this means that, with a better choice of  $CW$ , more stations could have been accepted. Otherwise, it means that our goal of maximizing the admissibility region is achieved.

Table 1 shows the results corresponding to  $R = 100\text{Kbps}$  and  $R = 200\text{Kbps}$  ( $\text{simu\_TH}$  refers to the simulated throughput obtained with the  $CW$  values given by our algorithm,  $\text{simu\_TH}_{max}$  to the maximum simulated throughput obtained from simulating all possible  $CW$  values, and  $\text{anal\_TH}$  to the analytical throughput calculated according to Eq. (12) with the  $CW$  values given by our algorithm). For  $n_{max}+1$  stations,  $\text{simu\_TH}_{max}$  is below the throughput requirement  $R$ , which leads to the conclusion that we achieve our goal.

### 4.3. Heterogeneous Throughput Requirements

In the above two simulations, homogeneous throughput requirements have been assumed. In order to validate our approach with heterogeneous throughput requirements,

Table 1  
Admissibility Region Test

$TH_{req}$		200Kbps	100Kbps
$n_{max}$		8	16
$n = n_{max}$	anal_TH	203.11Kbps	101.22Kbps
	simu_TH	202.05Kbps	100.46Kbps
$n = n_{max}+1$	anal_TH	180.41Kbps	95.25Kbps
	simu_TH <sub>max</sub>	180.78Kbps	95.39Kbps

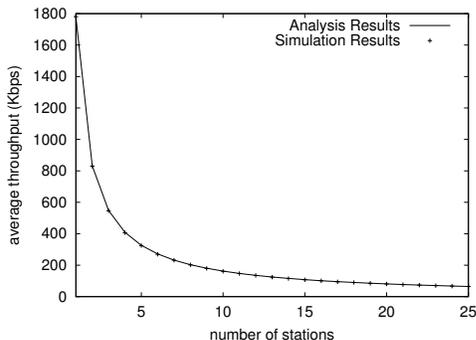


Figure 2. Simulation results with homogeneous throughput requirements.

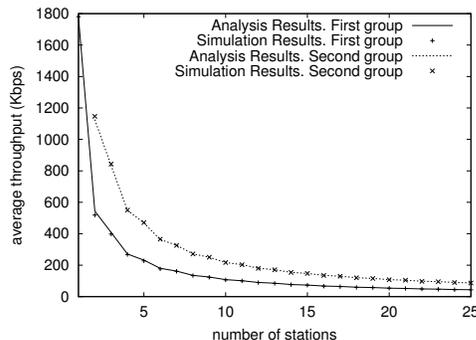


Figure 3. Simulation results with heterogeneous throughput requirements.

we repeated the simulation of Section 4.1 but with two groups of stations with different throughput requirements,  $R_1$  and  $R_2$ . Fig. 3 shows the analysis and simulation results corresponding to this new scenario when  $R_1 = 100\text{Kbps}$  and  $R_2 = 200\text{Kbps}$  (stations 1, 3, 5, ... have a throughput requirement of  $R_1$  and stations 2, 4, 6, ... have a throughput requirement of  $R_2$ ).

The maximum number of stations that our admission control can accept in this case is  $n_{max} = 11$  (6 stations with a throughput guarantee of  $R_1$  and 5 with  $R_2$ ). Simulation results show that for this value of  $n$ , the stations of the first group experience an average throughput larger than 100Kbps and the stations of the second group experience an average throughput larger than 200Kbps, which confirms that our algorithm also meets the desired throughput guarantees for the case of heterogeneous throughput requirements.

## 5. Conclusion

In this paper, we have presented an algorithm for providing throughput guarantee services in 802.11e Wireless LANs. This service matches the Assured Forwarding PHB proposed within the DiffServ architecture for the wireline Internet [4]. Since Wireless LANs may be considered as just another technology in the communication path, we argue that it is desirable that the architecture for QoS support follows the same principles in the wireless network as in the wireline Internet, assuring compatibility among the wireless

and the wireline parts.

From the analysis and simulation results, we have shown that our algorithm (1) provides the desired throughput guarantees to the accepted stations; and (2) within the level of freedom given by some restrictions we impose on account of simplicity, it maximizes the admissibility region.

Our approach is based on the 802.11e EDCF. The support of Best-Effort services within our architecture has been left as part of the future work. The support of real-time services should be better based on the 802.11e HCF and is out of the scope of our work.

The solution that we have proposed in this paper provides the desired throughput guarantees in absence of hidden nodes and channel errors. In presence of hidden nodes and errors, the service will degrade and the throughput guarantees will not be met. We argue that this feature is inherent in wireless media and the service we propose of guaranteed throughput in absence of impairments is the best that can be aimed at in such type of medium.

The 802.11 TGe is discussing the possibility of imposing some restrictions on the contention windows and the PFs used by the stations, specifically, forcing  $CW_{min}$  and  $CW_{max}$  to be multiples of 2 and fixing PF to 2. We believe that the flexibility to choose any value for  $CW_{min}$ ,  $CW_{max}$  and PF gives room for a more optimized operation, being the throughput guarantee services proposed in this paper a good example for this.

## REFERENCES

1. IEEE 802.11, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, Standard, IEEE, Aug. 1999.
2. C. Heegard, J. Coffey, S. Gummadi, P. Murphy, R. Provencio, E. J. Rossin, S. Schrum, and M.B. Shoemake, "High Performance Wireless Ethernet," *IEEE Communications Magazine*, vol. 39, no. 11, Nov. 2001.
3. IEEE 802.11e/D2.0, *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 – 1999 Edition, 2001.
4. J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski, "Assured Forwarding PHB Group," RFC 2597, Jun. 1999.
5. A. Imad and C. Castelluccia, "Differentiation Mechanisms for IEEE 802.11," in *Proc. IEEE INFOCOM'01*, Anchorage, Alaska, Apr. 2001.
6. T. S. Ho and K. C. Chen, "Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LAN's," in *Proceedings of IEEE PIMRC*, Taipei, Taiwan, Oct. 1996, pp. 392–396.
7. Giuseppe Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, Mar. 2000.
8. "The Network Simulator – ns-2," <http://www.isi.edu/nsnam/ns/>, Online Link.
9. A. Banchs and X. Pérez, "Providing Throughput Guarantees in IEEE 802.11 Wireless LAN," in *Proc. WCNC'02*, Orlando, FL, Mar. 2002.