

Dynamic EDCF Parameters for QoS Control in IEEE 802.11 Hot Spots

José Coelho de Melo Filho, Luci Pirmez¹ and José Ferreira de Rezende²

¹ Núcleo de Computação Eletrônica - Universidade Federal do Rio de Janeiro

CCMN Bloco C - CP 2324 - 20001-970 - Rio de Janeiro RJ - Brazil

jcoelho, luci@nce.ufrj.br

² Grupo de Teleinformática e Automação - Universidade Federal do Rio de Janeiro

PEE-COPPE - CP 68504 - 21945-970 - Rio de Janeiro RJ - Brazil

rezende@gta.ufrj.br

Abstract

This paper describes a control mechanism for dynamic selection of Quality of Service (QoS) parameters used in a IEEE 802.11e Wireless Local Area Network (WLAN). The 802.11e specification, still under development, provides a new contention-based access method, called Enhanced Distributed Coordination Function (EDCF). This method uses distinct values for interframe space and for contention window size to establish priorities between Access Categories (ACs). The proposed mechanism selects these values based on the actual network load condition and on the applications delay requirements. Such mechanism improves the new QoS functionality, aiming to promote better channel utilization as well as to keep the desired service levels.

Keywords: WLAN, IEEE 802.11, Quality of Service.

1 Introduction

The IEEE 802.11 standard for Wireless LANs [1] will play a significant role in the wireless Internet access scenario. It's a high-rate, low-cost and license-free technology, which achieved a high degree of maturity and popularity. 802.11 Access Points (APs) can be easily found in many countries around the world. They have been installed at university *campi* and at places with high user concentration such as hotels, airports and conference halls. The standard is a strong candidate to make part of a wide-coverage access system, either integrated to third-generation cellular networks or simply composing a whole 802.11 solution for home networking and hotspots.

However, the limited support for applications with Quality of Service (QoS) requirements is one of the major 802.11 shortcomings. To address this issue, the IEEE is developing the 802.11e extension [9], named "MAC Enhancements for Quality of Service"¹. This new specification extends the protocol in terms of QoS functionalities and improves its capabilities and efficiency. The 802.11e extension adds two new access methods in the medium access control (MAC) sub-layer, one contention-based and another contentionless, both with QoS functionalities. These new methods allow the application of differentiated treatment for traffic categories with distinct QoS requirements. In the 802.11e extension a QoS-aware coordinator, called Hybrid Coordinator (HC), is able to issue, included in some control frames, an information called QoS Parameter Set, which is used by QoS-aware stations to configure the MAC parameters of their wireless interfaces. This parameter setting affects the medium access behavior for each traffic class inside a station. This way, the coordinator is able to change the traffic load according to differentiation rules and access priorities.

Although the 802.11e extension provides the tuning knobs for the QoS control, it does not define how this support should be used given specific network conditions. This motivates the development of new mechanisms, concerning the monitoring and control of service levels, which complement the QoS functionality introduced by the 802.11e extension. Such mechanisms are important to guarantee the efficient use of the available network resources and to allow the establishment and maintenance of adequate QoS levels to the existing applications. An open issue in the new

¹It is still a work in progress by the time of the present writing.

contention-based access method, for example, is how the QoS Parameter Set should be composed for each particular network condition and when it should be changed.

The work described in this paper is concerned with the development and evaluation of one of these QoS control mechanisms, enabling 802.11e APs (QAPs) to provide the desired service differentiation, while maintaining high network utilization levels. To achieve this goal, we develop a rate control mechanism for dynamic selection of the QoS parameters informed in the QoS Parameter Set. We evaluate the mechanism using admission-controlled UDP real-time traffic in conjunction with TCP best effort traffic. The AP performs the rate control function besides the provision of interconnection to wired networks for a set of wireless stations (STAs). The rate control is intended to regulate the network load imposed by best effort sources and to force them to defer access to real-time traffic. It was also designed to allow best effort sources to exploit as much as possible bandwidth left available by QoS-restricted ones. The rate control algorithm is based on the network load condition and performance metrics measured by the AP at regular intervals. It affects all data sources associated to a traffic class or category instead of individual sources. The AP uses only locally available information to undertake control decisions. No feedback about flows is used between the STAs and the AP. We only assume that a mechanism, like the QoS Parameter Set itself, is available to inform the STAs about the MAC parameters to be used in each situation. Issues related to the QoS control functionality in the contentionless access method [7] are out of scope of this work.

This paper is structured as follows. Section 2 presents basic concepts of 802.11 networks. Section 3 relates some service differentiation techniques developed for the 802.11 MAC. In Section 4, the IEEE 802.11e extension is presented. Section 5 describes the rate control mechanism implemented. Section 6 shows the results of simulations performed to evaluate the mechanism. Section 7 lists some related work. Finally, Section 8 presents the conclusions.

2 Background on IEEE 802.11

The Basic Service Set (BSS) is the building block of the 802.11 architecture [1]. A BSS is defined as a group of stations (STAs) that are under direct control of a coordination function. The coordination function determines when a STA can transmit and receive data in a BSS. In the infrastructure mode, in contrast to the

ad hoc mode, the AP establishes the communication among the STAs themselves and the wired network. It also provides other functionalities such as mobility and security management.

The 802.11 defines two types of coordination functions to control the medium access: the Distributed Coordination Function (DCF) and the Point Coordination Function (PCF). In the PCF mode, a Point Coordinator, which resides in the AP, controls the access to the medium through periodic polling of each STA, which transmits without contention. The DCF provides a medium access with contention and uses a multiple access protocol with carrier detection (CSMA) similar to CSMA/CD used in the IEEE 802.3 networks (Ethernet). To allow coexistence of PCF and DCF in the same BSS, Inter Frame Spaces (IFS) are introduced in each mode, called *PIFS* (Point Inter Frame Space) and *DIFS* (Distributed Inter Frame Space), respectively. Therefore, STAs in the DCF mode have to sense the medium free for a period of time *DIFS* before trying to contend for it. The Point Coordinator waits for a period of time *PIFS* before issuing a control frame that announce the beginning of a contention-free period. In this period, the AP polls the PCF STAs while the DCF ones remain silent. As *PIFS* is smaller than *DIFS*, the AP always has priority over the DCF STAs to start a contention-free period. To avoid this to occur at any instant, a smallest period of time *SIFS* (Short Inter Frame Space) is used between the frames. The contention period follows the contention-free period forming what is called a super-frame.

In the DCF mode (figure 1), when a STA decides to transmit and senses the medium is busy, it postpones the transmission. A persistent or non-persistent mode may be used. The foremost difference for the CSMA/CD used in wired networks is that collision detection is not possible in wireless networks due to radio limitations [11]. This function is replaced by mechanisms of collision avoidance and positive acknowledgments (ACKs) of transmitted frames. The method is known as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). In this method, receiving STAs must acknowledge each received frame what guarantees that a collision did not occur. Besides, STAs backoff for random intervals (backoff timer) prior transmitting frames in the attempt of avoiding collisions. The backoff timer is randomly chosen within the interval that goes from zero to the current contention window size (*CW*). The *CW* assumes a minimum size (*CW_{min}*) in the beginning of the communication and after each successful trans-

mission. In order to prevent stations from capturing the medium, stations that loose the contention (larger backoff timers) suspend their backoff timers when they detect the medium is busy. This residual backoff will be used for these stations in the next contention for the medium. Stations that backoff for the same interval, causing a collision, have their contention window increased to the next power of 2 minus 1 until a maximum value is reached (CW_{max}). This reduces the probability of repeated collisions among STAs trying to transmit. If an acknowledgement frame (ACK) is not received before a timer expires, the transmitting STA also doubles its contention window. To prevent the hidden terminal problem, an optional access method is defined where STAs exchange RTS/CTS frames prior to transmitting data frames.

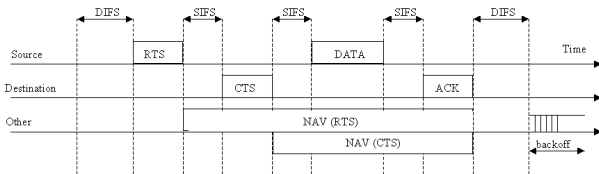


Figure 1: DCF using RTS/CTS.

3 Priority-Based QoS Support

Service differentiation in the 802.11 MAC sublayer has been widely studied. Some techniques have been elaborated for both access methods, with contention (DCF) and without contention (PCF). Performance analysis of techniques applied in the DCF access method have been done [2, 4, 6, 12, 13]. These techniques generally involve one or more basic parameters of MAC 802.11. They modify the way each STA accesses the medium. The distinct configuration of contention window size ($[CW_{min}:CW_{Max}]$), $DIFS$, frame size, and RTS Threshold establishes access priorities between STAs. $DIFS$ and backoff affect the time STAs wait before trying to access the medium, while the frame size regulates the time a STA retains it after winning a contention. The smaller the $DIFS$ or CW used by a STA the higher its priority. Similarly, a larger frame size gives higher throughput for a certain STA while impacts the average delay experienced by the others. However, larger frames are more prone to transmission errors, what reduces the effectiveness of this mechanism in channels with high error

rates [2]. The $DIFS$ and backoff techniques are similar, but the differentiation in each case is deterministic and statistic, respectively.

These techniques can be combined to compose more complex solutions. They can be applied to establish not only a station-based differentiation, but also per flow (or per traffic category) and even per packet differentiation [2]. As an example of per-packet differentiation, the work in [8] defines a distributed priority scheduling scheme where the priority of the next packet in the head of the queue of each STA is informed to the others by means of RTS, CTS, data and ACK frames. The priority rank of each data frame establishes the IFS and/or backoff interval to be used prior the transmission of this frame. Additional information on 802.11 MAC differentiation and other MAC related problems can be found in [2, 4, 12, 13].

MAC 802.11 presents limitations with respect to the fairness in the bandwidth sharing and to the performance and support for real-time applications. These problems, in conjunction with the imperative need of QoS provisioning in WLANs, have pointed to the necessity of evolution of the 802.11 standard.

4 The 802.11e Extension

Due to the recent developments in the QoS support for 802.11 networks and the consequent need for standardization, the IEEE created a working group (802.11e) that is specifying "MAC Enhancements for Quality of Service" to these networks. In the upcoming 802.11e standard, the STAs are called QSTAs (QoS Stations), the access point is the QAP (QoS Access Point), and the BSS is named QBSS (QoS Basic Service Set). The extension incorporates a new coordination function called Hybrid Coordination Function (HCF). In this function, a Hybrid Coordinator (HC), normally located in the QAP, is responsible for the establishment and coordination of the QoS functionality within the QBSS. The hybrid function combines characteristics from the PCF and the DCF and allows a transparent coexistence with legacy stations (802.11/b/a).

A new important concept in 802.11e is the Transmission Opportunity (TxOP). A TxOP is defined as a time interval in which a STA has the right to initiate transmissions. It is defined by a beginning time and duration interval. During a TxOP, a STA can send multiple frames in bursts, separated by SIFS, without having to contend for the medium at each frame. The duration of a TxOP is bounded to prevent large access delays in STAs waiting to access the medium. The

HCF function also provides two access methods: a polling-based, called HCF polled channel access, which is centralized in the HC; and a contention-based, called HCF contention-based channel access, which is executed distributed in the STAs.

In the polling-based access, TxOPs are granted explicitly by the HC (polled TxOPs) following scheduling schemes [7]. Polled TxOPs can be used at any instant to satisfy priority deliveries, throughput, delay and jitter requirements. The HC creates periods of controlled access with limited duration (CAPs - Controlled Access Periods) using its privileged access and knowledge of existing flows in the QBSS.

In the contention-based access, QSTAs compete for TxOPs in a distributed way using an Enhanced DCF access mode or EDCF. The EDCF incorporate multiple transmission queues in the same STA, each one with a priority level. The mechanism is based on a previous proposal called Virtual-DCF or VDCF [5, 6]. Within each QSTA, DCF instances are executed in parallel with different prioritized queues and MAC parameters (Figure 2). These instances, defined as Access Categories (ACs), contend externally (with other stations trying to transmit) and internally (with other ACs supported by the QSTA). Internal (virtual) collisions are solved granting the access to the high-priority AC, while low-priority ACs enter backoff, alike is done in external (physical) collisions. Each 802.11e QSTA can support from 2 to 8 ACs.

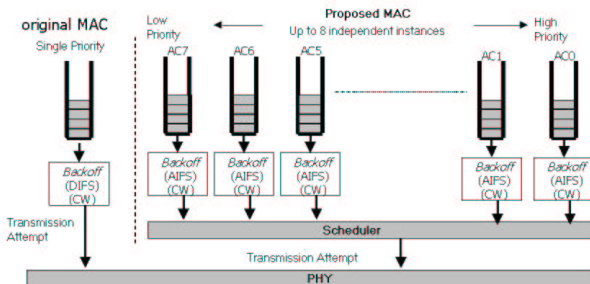


Figure 2: The New 802.11e MAC.

The IFS of each AC is called Arbitration Interframe Space (AIFS). High-priority categories use smaller values for *AIFS*. The QoS parameters are defined by the HC based on 8 User Priorities (UPs). The QoS Parameter Set (or EDCF Parameter Set) is used to inform the QSTAs about MAC instances configuration parameters. The EDCF Parameter Set gives the value of *AIFS(i)* and *CWMin(i)* being used by the priorities

from 0 to 7. *AIFS* and *CWmin* fields have 8 octets; each octet represents an unsigned integer that indicates the value of *AIFS* or *CWmin* for each priority *i*. It is transmitted by the HC encapsulated in control frames as Beacon and Probe Response [1]. The maximum duration of the transmission after the contention is defined by the TxOP Limit parameter. It is informed in a 2 octets field that indicates a number of periods of 16 μ s corresponding to the TxOP maximum duration. In contrast to the *AIFS(i)* and *CWMin(i)*, this value is valid for all the QBSS and not per AC.

5 Rate Control Mechanism

In this paper, values transmitted in the EDCF Parameter Set are defined by a rate control mechanism. This mechanism monitors periodically the network load, collecting statistics from the AC queues within the QAP, such as average load and data frames access delay observed within a monitoring interval. Based on these statistics, the algorithm modifies the *AIFS* and *CWmin* values of each category aiming to bound the access delay for high-priority categories, while increasing the network use level by low-priority ones.

5.1 Effect of *AIFS* and *CWmin* Parameters

Before we develop a rate control based on *AIFS* and *CWMin* changes, the effects of these parameters had to be studied. Although *AIFS* and *CWmin* control the STA waiting time (access delay), some differences between these parameters exist that should be considered. The *AIFS* controls a deterministic part of this time, while the *CWmin* controls a probabilistic one. The consequence of this differences to the rate control is that changes of *AIFS* value cause an immediate answer from the STA, while *CWmin* changes are only perceivable after a certain time interval. The *CWmin* also affects the average access delay of a source more aggressively than the *AIFS* since the former parameter only assumes values that are power of 2 minus 1. With *AIFS*, it is possible to add one slot at a time to control the access delay, what allows a less aggressive control.

Some experiments were performed during the present work to evidence the effects of these two parameters. Figures 3 and 4 show the effect of the *AIFS* on the aggregate throughput obtained by one and eight CBR sources, respectively. The IFS interval of each category is defined by $SIFS + AIFS(i) \cdot SlotTime$, where *AIFS(i)* gives the number of slots. The CBR

sources are assigned to the same AC. All QSTAs belong to the same QBSS and use a maximum rate of 11 Mbps. In the two figures, each curve shows the aggregate throughput by the addition of sources in an isolated experiment. In each experiment the packet size (p) varies from 64 to 2304 bytes. Observing Figures 3 and 4, one can notice that an increasing number of slots in the *AIFS* causes a linear reduction in the aggregate throughput, independently of the number of sources in the BSS.

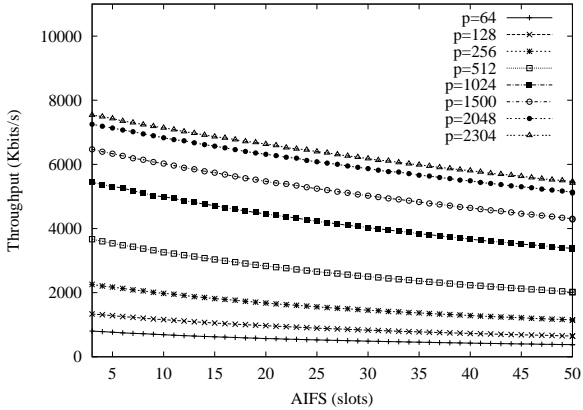


Figure 3: Throughput for 1 CBR Source x AIFS Size.

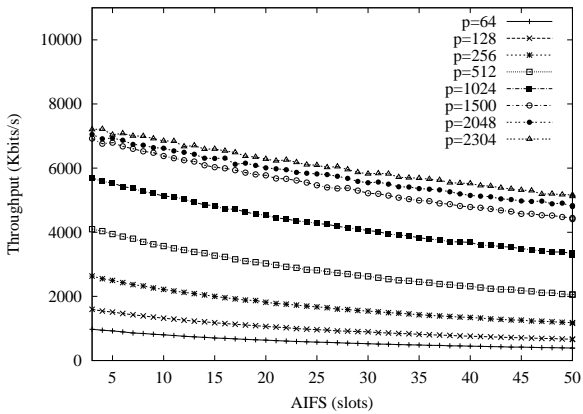


Figure 4: Throughput for 8 CBR Sources x AIFS Size.

Figures 5 and 6 show the same previous experiments as function of the CW_{min} variation. The CW_{min} values vary from 31 to 4095, becoming periodically $CW_{min} = (CW_{min} * 2) + 1$. The AIFS is kept constant and equal to 2.

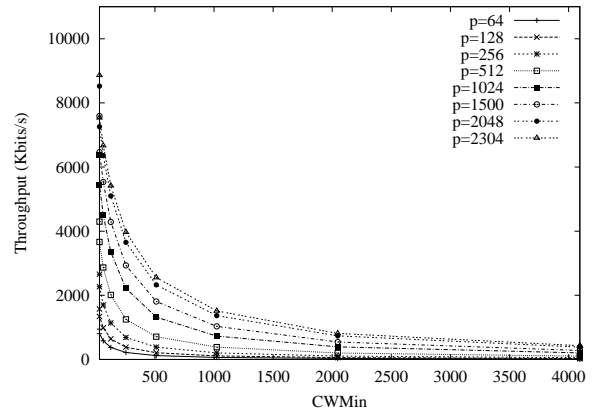


Figure 5: Throughput for 1 CBR Source x CWMin Size.

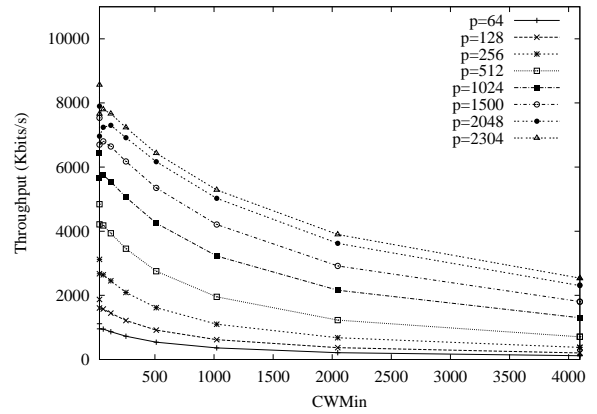


Figure 6: Throughput for 8 CBR Sources x CWMin Size.

While for only one source the exponential increase of CW_{min} provokes a great reduction in the obtained throughput, for 8 sources, however, a different behavior is observed, i.e. the aggregate throughput is more immune to the variations of CW_{min} . This can be explained considering the importance that the contention window size has in the collision avoidance. In overload situations, as explored in these experiments, the collision probability increases with the number of MAC instances that contends for the medium. Each instance that loses a transmission opportunity has to enter in backoff using a larger value of CW. The average time spent in backoff until it obtains a successful transmission is then equal to

$\frac{1}{2} \left[\sum_{i=1}^t (32^i - 1) \right] \cdot SlotTime$ where t is the average number of transmission attempts, which increases with the average number of necessary attempts for a successful transmission. Increasing the $CWmin$ value decreases the probability that two instances choose the same slot, enlarging the possibility of transmission in the first attempt and the subsequent ones. The aggregate throughput only starts to decrease as function of the $CWmin$ increase when the value of this parameter becomes extreme large in relation to the ideal value for collision prevention.

5.2 Restrictions

Given the differences on the effects of $AIFS$ and $CWmin$, the variations of these two parameters can be used in conjunction, as complementary control resources. The rate control uses $AIFS$ to define the separation degree between priority levels, while the $CWmin$ is modified according to the number of EDCA instances competing in one certain category. However, some restrictions must be observed in the choice of these values.

- Restriction 1: $AIFS(i) \geq 2, \forall i$, to prevent malfunctioning of the MAC sublayer. $AIFS(i)$ cannot be 0 to prevent it being equal to SIFS and it cannot be 1 since this value is used by PIFS, defined by $SIFS + SlotTime$.
- Restriction 2: $AIFS(i+1) \geq AIFS(i)$ in order to preserve the differentiation. If $AIFS(i+1) = AIFS(i)$, then forcibly $CW(i+1) > CW(i)$. This allows that low-priority categories explore the maximum channel capacity and keeps the possibility of differentiation in case the activity in high-priority categories restarts.
- Restriction 3: $AIFS(i+1) < AIFS(i) + CW(i)$ if the load of category i is high, to avoid the starvation effect on category $i+1$. Such effect occurs whenever the lower priority category have no opportunity to transmit its frames while a higher priority category is transmitting.
- Restriction 4: $CWMin(i) < CWMax, \forall i$, as specified in the 802.11 standard.
- Restriction 5: $CWMin \geq 7$ for the high-priority category, considering that this value is a reasonable minimum to allow a trade-off between delay and collision avoidance. For the best effort category, $CWMin \geq 31$, in order to prevent priority over legacy stations (802.11/11b/11a).

In general, a category i cannot have smaller $CWmin$ than $2^{3+i} - 1$, so that restriction 2 is also respected.

5.3 Algorithm

The rate control is executed periodically in accordance with a monitoring interval. To each execution, the control verifies if the used $CWmin$ in the categories is adjusted, using the `fixContention(i)` procedure, from the highest priority. After that, it performs the adjustment of the load imposed by the lower priority categories, increasing or decreasing the $AIFS$ and $CWmin$ values of these categories, always in view of the restrictions and the objectives of differentiation, as bounded delay or minimum throughput in one determined category. After the execution of these two operations in all categories, the control issues a new EDCA parameter set.

Figures 7, 8 and 9 show the algorithms used in the specific case of two categories with the coexistence of real-time and best effort traffic. In Figure 7 is shown the main procedure of the control. In the case of only two categories, it calls the `fixContention` procedure for the high-priority category, shown in Figure 8. In this procedure, it is verified if the $CWmin$ of the high-priority category should be larger to reduce the number of transmission attempts. This is done comparing the value of $CWmin$ with the number of accepted real-time sources in the category. This information is obtained from the admission control of real-time traffic².

```

controlRate() /* executed each monitoring interval */
{
    fixContention(i)
    adjustLoad(i+1)
    issueQoSParameterSet()
}

```

Figure 7: Rate Control Main Procedure

A simple rule is used to decide whether the $CWmin$ value should be modified or not. If the number of accepted sources is greater than the number of slots in the $CWmin$, then $CWmin$ must be larger. With this simplification, we keep the probability of choice of one slot always next to $1/AcceptedSources$.

The procedure also verifies if $CWmin$ can be reduced, comparing the half of its value with the number of accepted sources. In case the $CWmin$ size of

²The real-time traffic must be subject to some sort of admission control to prevent that the maximum channel capacity is exhausted by this traffic category.

```

fixContention(i) /* adjusts CWMin of category "i" according
to the number of EDCF instances in this category */
{
  OldCW = CWMin(i)
  if AcceptedSources > CWMin(i) /* AcceptedSources
comes from Admission Control */
    CWMin(i) = CWMin(i) * 2 + 1
  if AcceptedSources < CWMin(i)/2
    CWMin(i) = (CWMin(i) - 1) / 2
  if CWMin(i) < 2**(3+i) - 1
    CWMin(i) = 2**(3+i) - 1 /* restriction 5:
CWMin=7 for i=0 */
  if CWMin(i) > OldCW /* CWMin has increased */
    AIFS(i+1) = AIFS(i+1) +
[(CWMin(i) - OldCW) * δ]
  if CWMin(i) < OldCW /* CWMin has decreased */
    AIFS(i+1) = AIFS(i+1) -
[(OldCW - CWMin(i)) * δ]
  if AIFS(i+1) < 2
    AIFS(i+1) = 2 /* restriction 1 */
}

```

Figure 8: fixContention Procedure

the real-time category is modified, the deterministic part of the waiting time of the best effort category will be increased or decreased. This is done by adding to $AIFS(i+1)$ the difference between the old $CWmin$ and the new one from the category i . If all the difference is added to the $AIFS(i+1)$, the size of the overlapping region of the waiting time probabilistic parts for the categories i and $i+1$ is kept constant in number of slots. It is possible to use only one fraction of this difference by multiplying it by a factor δ that varies from 0 to 1.

After that, the procedure $adjustLoad(i+1)$ (Figure 9) is executed. It verifies whether the best effort load should be increased or reduced. If the delay suffered by the real-time category is above a maximum threshold, the best effort load must be reduced. If it is below an inferior threshold, what represents a condition of over provisioned resources for the real-time traffic, the load of the best effort can be increased. To prevent instabilities, an additional condition is placed in the load increase. The load can increase only if the current load is less than or equal to the previous load minus the average load of a single source, suggesting (but not assuring) that one real-time source left from the previous period to the current one.

The load reduction is performed initially adding a certain number of slots (*ReductionSlots*) to $AIFS$ up to the limit of restriction 3. Above this point, $CWmin$ is doubled up to the limit of restriction 4. The increase is made in the inverse way, respecting restrictions 1 and 5 and using a different number of slots

```

adjustLoad(i+1) /* verifies the QoS threshold */
{
  if PresentDelay(i) > MaximumDelay(i)
    reduceLoad(i+1)
  else if (PresentDelay(i) <
MinimumDelay(i))e(PresentLoad(i) <=
PreviousLoad(i) - SourceMeanLoad(i))
    increaseLoad(i+1)
}
reduceLoad(i+1) /* reduces the load of category i */
{
  if AIFS(i+1) < (AIFS(i) + CWMin(i)) /*
restriction 3 */
    AIFS(i+1) = AIFS(i+1) + ReductionSlots
  else if CWMin(i+1) < CWMax /* restriction 4 */
    CWMin(i+1) = (CWMin(i+1) * 2) + 1
}
increaseLoad(i+1) /* increases the load of category i */
{
  if CWMin(i+1) > 31 /* restriction 5 */
    CWMin(i+1) = (CWMin(i+1) - 1) / 2
  else if AIFS(i+1) > 2 /* restriction 1 */
    AIFS(i+1) = AIFS(i+1) - IncrementSlots
}
}

```

Figure 9: adjustLoad Procedure

(*IncrementSlots*), lower than *ReductionSlots*. Using a lower number of slots turns the control less aggressive in the load increase than in the decrease, what it is good for the mechanism stability. After the execution of these procedures for all categories, a new QoS parameter set is composed and transmitted to the QSTAs.

6 Simulation

The simulation tool used in the performance evaluation was the Network Simulator (NS) version 2.1b8a. In order to build 802.11 infrastructure-mode networks in this tool, the routing agent NOAH (NOn Ad Hoc) is used. Modifications in the 802.11 MAC code have been necessary so as to support the EDCF functionalities. The multiple queues implementation had as starting point the Virtual DCF code [5]. The rate control performance evaluation explores the co-existence of real-time and best effort traffic in the same QBSS.

The simulation scenario consists of a QAP connected to a fixed node by a 100 Mbps link with 2 ms of propagation delay. All 802.11 stations and the QAP have a transmission range of 250 meters. The QAP is located in the center of an area of 350x350 meters, in such a way that all mobile stations inside this area are within the transmission range of the QAP. The data rate is of 11 Mbps and the basic rate is of 1 Mbps. The scenario, shown in Figure 10, simulates a 802.11 hotspot providing access to the wired network.

The simulations are run using the basic access scheme, i.e. without the use of RTS/CTS.

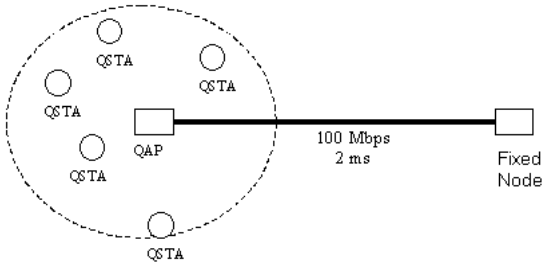


Figure 10: Scenario Used in Simulations.

The real-time traffic is composed by voice traffic modeled by on-off sources which have activity and silence periods following exponential distributions with averages 1.2 and 1.8 s, respectively. These sources generate 64 kbps during the activity period and use packets of 210 bytes³ in a UDP connection. The arrival interval of voice sources is uniformly distributed between 0 and 7 s. To simulate the conversational behavior of voice connections, two sources are used: one from the fixed node to the mobile nodes (downlink) and another from the mobile nodes to the fixed node (uplink). By using a moving time-window admission control, the maximum allowed number of simultaneous voice sources in the QBSS is around 25.

The best effort traffic is generated by long-term TCP connections using eight FTP sources that continuously transfer data in 576-byte packets from the fixed node to the mobile nodes. The voice traffic uses the higher priority access category (AC0) and the FTP traffic uses the lower priority category (AC1). The AC0 uses $AIFS=2$ and $CWmin=7$ and the AC1 initiates with $AIFS=9$ and $CWmin=31$.

To observe the effect of the arrival and departure of voice sources on the throughput obtained by the FTP traffic, the voice sources are admitted up to 150 s of simulation and have a fixed duration of 250 s. This way we can observe 4 phases in the simulation run: the admission phase (phase 1), between 0 and 150 s; the steady phase (phase 2) between 150 and 250 s, where the maximum number of voice sources is present; the departure phase (phase 3), from 250 s until 400 s (depending on the admittance time of the last source); and, the FTP-only phase (phase 4), where only FTP sources and one voice source are present. The first

³This model approximates a PCM (Pulse Code Modulation) coder with silence suppression.

admitted voice source is active during the whole simulation, and it is used to monitor the uplink traffic delay. All simulation runs last for 600 s.

To establish a comparison basis for the rate control mechanism performance, simulations with and without the use of the control have been run. In the simulations with rate control, the monitoring interval is of 3 s, what is equivalent to an average talk-silence cycle. The maximum and minimum access delay for the voice packets, used in the `adjustLoad()`, are of 20 ms and 4 ms, respectively⁴. In this procedure, 1 slot is used for increase and 4 slots for reduction. The parameter δ used in the `fixContention()` procedure is equal to 0.8.

6.1 Numerical Results

The results with and without rate control were collected in 10 simulation runs of the scenario. In each run, we measure the aggregate throughput of FTP traffic (average during all the simulation) and the average access delay of all voice packets that enter the transmission queue of the AP in the downlink direction.

Tables 1 and 2 show the access delay of voice frames in the downlink direction for cases without and with rate control, respectively. In these tables, the column "delay" shows the average delay and the 95% confidence interval measured over 10 runs. The columns "P-90", "P-95" and "P-99" show the 90th, 95th and 99th delay percentiles, respectively.

Table 1: Delay of Voice Frames without Rate Control.

	delay (ms)	P-90	P-95	P-99
mean	105.60	4.26	14.63	123.59
CI-95	44.84	1.06	7.97	68.90

Table 2: Delay of Voice Frames with Rate Control.

	delay (ms)	P-90	P-95	P-99
mean	4.20	1.82	3.07	7.38
CI-95	1.53	0.15	0.24	2.01

The results demonstrate that the rate control contributed to keep the access delay bounded, below the 20 ms proposed. Both the average delay and the 99th percentile of delay are below the maximum threshold

⁴The maximum delay was chosen by purpose inferior to the packet generation interval of the voice application.

in all 10 runs. This did not occur without the rate control. The static definition of $AIFS$ and $CWmin$ values in the presence of best effort traffic revealed to be insufficient to guarantee any control over delay bounds of the real-time traffic. If larger $AIFS$ and $CWmin$ values for the FTP category (AC1) were used, it would be possible to obtain smaller values for delay of the voice traffic without rate control, but the FTP throughput would be excessively reduced.

On the other hand, the rate control mechanism makes possible to keep the delay bounded without excessively sacrificing the best effort traffic. Therefore, the mechanism reduces the FTP load only in the most critical instants for the voice traffic, and increases it always that is possible. Table 3 shows the aggregate average throughput of FTP traffic. The "average" column shows the average throughput in 10 runs (with and without rate control) and the "CI-95" column discloses the 95% confidence interval. The first line shows measures considering all the simulation, and the remaining four show the results for each one of the previously cited phases.

Table 3: FTP Traffic Average Throughput.

Period	Without Rate Control		With Rate Control	
	mean (Kbps)	CI-95	mean (Kbps)	CI-95
0 a 600 s	1319.23	24.69	1206.54	47.32
Phase 1	1025.74	36.96	816.75	40.29
Phase 2	625.47	51.40	341.05	46.51
Phase 3	1609.62	36.33	1533.26	125.87
Phase 4	2010.43	11.72	2186.46	33.83

In the first line of this table the period of simulation is observed that the control provoked a reduction of only 8.5% in the best effort traffic throughput. This significant result is similar to that reached for Ahn *et al* in the SWAN model [3], described in Section 7, where a delay 60-75% minor for the real-time traffic is obtained with the reduction of only 2% in the throughput of the best effort traffic.

In phases 1, 2 and 3 we can verify that the FTP throughput was always inferior to the case with rate control, especially in phase 2, when a maximum number of voice sources is present in the QBSS. On the other hand, in phase 4 the FTP throughput is bigger in the case with rate control, because the `adjustLoad()` procedure recognizes that there is fewer voice traffic and it is able to select inferior values for $AIFS$ and $CWmin$ until $AIFS=2$ and $CWmin=31$ (default values in a 802.11b network).

Figures 11 and 12 show the maximum access delay

value (P-100) of voice packets in two specific simulation runs, without and with rate control. The values are collected periodically in the QAP by monitoring the AC(0) transmission queue. One observes that, without rate control, the expected maximum delay bound (20 ms) is violated a number of times between 100 and 300 seconds of simulation, attaining peaks superior to 600 ms. This is critical for voice traffic, considering that in this kind of application an end-to-end delay of 150 ms should be guaranteed for best quality service. The rate control of low-priority ACs is able to confine statistically the access delay of high-priority ACs between a minimum and maximum thresholds.

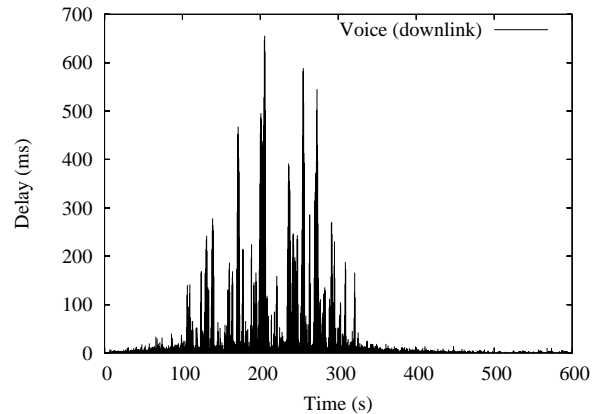


Figure 11: Access Delay of Voice Traffic without Rate Control for a Particular Simulation Run.

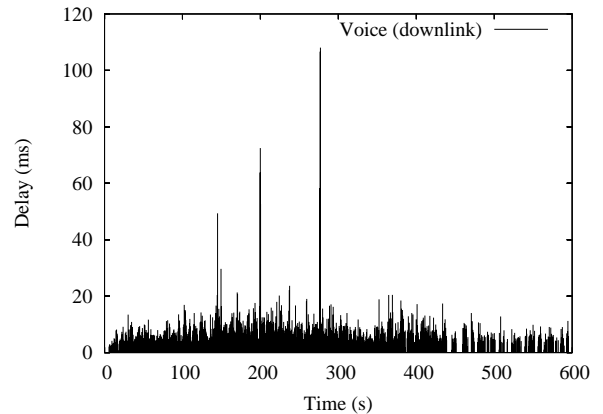


Figure 12: Access Delay of Voice Traffic with Rate Control for a Particular Simulation Run.

7 Related Work

At our knowledge, the closest to the present work is the SWAN (Stateless Ad hoc Wireless Networks) network model [3]. In that model, among other functionalities, the authors propose an admission control for soft real-time traffic and a rate control for best effort traffic in 802.11 networks. However, there are some differences between the rate control mechanism used here and that of the SWAN model. First, the SWAN model is intended for ad-hoc networks, where an infrastructure to access fixed networks is not available. Second, the SWAN conception assumes no modification in the MAC sublayer while we suppose the existence of QoS-aware nodes. Third, the SWAN rate control is performed by an additive increase multiplicative decrease (AIMD) algorithm in the parameterization of token buckets acting as traffic source shapers. In contrast, we perform rate control indirectly by changing MAC parameters and, hence, altering the maximum throughput that a best effort source can attain.

Romdhani et. al introduced the Adaptive EDCF (AEDCF) [10] that dynamically adapts the MAC parameters according to the network load conditions. This scheme also uses the new access method with contention of the 802.11e (EDCF). The AEDCF scheme, however, was also idealized for ad-hoc networks and for distributed computation.

Given the conceptual differences, these approaches are not simply comparable. However, we believe that decentralized approaches such as the SWAN and AEDCF are less suitable to the wireless access scenario than a centralized one. They add computational complexity to the mobile stations that is not desirable when support to low cost stations is intended. For this reason, we try to keep the complexity restricted to the AP. We also try to take advantage of some new 802.11e features combining them with traditional control schemes adapted to the wireless scenario. Thus, our QoS control explores service differentiation without compromising the efficiency in the use of the channel and without adding complexity to the STAs, requiring only the compatibility with the 802.11e extension.

8 Conclusions

This article described a rate control mechanism that selects differentiation parameters used in the distributed coordination function (DCF) of the 802.11e standard, in accordance with the network load condition and the applications requirements. This me-

chanism was evaluated by simulations that explored scenarios of coexistence of real-time and best effort traffic.

The mechanism shows that it is feasible to confine the access delay observed by real-time traffic frames between a maximum and a minimum threshold, suggesting that a parameterized QoS is possible using the new contention-based access mode of the 802.11 standard. Moreover, it allows the best effort traffic to explore the maximum of the bandwidth not used by the real-time traffic. The results obtained are compatible with the results of others works [3, 10], and certify that a dynamic selection of the differentiation parameters is preferable in relation to a static one. Therefore, it allows a better delay control for real-time traffic in high load situations and a greater channel use for best effort traffic when low real-time traffic is present in the network.

The mechanism has the advantage to keep the complexity of QoS control restricted to the 802.11e QAP, what can be important when is intended to support low cost mobile stations. The presented algorithms can be extended for more than two access categories. The conditions used for adjusting contention and load can vary, depending on the particular QoS objective that one wants to guarantee using the rate control.

Future works include experiments using a greater dynamic of source arrival and departure, as well as other source models for real-time and best effort traffics. It is also intended to extend the mechanism for more than two access categories, to study the interactions of 802.11e QoS support with the QoS architectures for wired networks and to develop other mechanisms that complement the new QoS functionality created for the wireless local area networks.

Acknowledgments

This work has been supported by UFRJ, FAPERJ, CNPq, CAPES, and COFECUB. We also want to thank Mr. Greg Chesson and Mr. Aman Singla for developing and making available the V-DCF code.

References

- [1] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Standard 802.11, 1999.

- [2] I. Aad and C. Castelluccia. Differentiation Mechanisms for IEEE 802.11. In *Infocom'2001*, Anchorage, Alaska, April 2001.
- [3] Gahng-Seop Ahn, Andrew T. Campbell, Andras Veres, and Li-Hsiang Sun. Service Differentiation in Stateless Wireless Ad Hoc Networks. In *INFOCOM'2002*, New York, USA, June 2002.
- [4] Brahim Bensaou, Yu Wang, and Chi Chung Ko. Fair Medium Access in 802.11 Based Wireless Ad-Hoc Networks. In *IEEE/ACM The First Annual Workshop on Mobile Ad Hoc Networking & Computing (MobiHoc'00)*, Boston, USA, August 2000.
- [5] Greg Chesson, Wim Diepstraten, Maarten Hoebe, Aman Singla, Harold Teunissen, and Menzo Wentink. VDCF Presentation - IEEE Document 802.11-01/132. Technical report, IEEE 802.11 TGe, March 2001.
- [6] Greg Chesson, Wim Diepstraten, Duncan Kitchin, Harold Teunissen, and Menzo Wentink. Baseline D-QoS Proposal - IEEE Document 802.11-00/399. Technical report, IEEE 802.11 TGe, 2000.
- [7] Antonio Grilo, Mario Macedo, and Mario Nunes. A Scheduling Algorithm for QoS Support in IEEE 802.11e Networks. *IEEE Wireless Communications*, 10(3), June 2003.
- [8] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly. Distributed Multi-Hop Scheduling and Medium Access with Delay and Throughput Constraints. In *ACM The Seventh Annual Conference on Mobile Computing and Networking (MobiCom'01)*, Rome, Italy, July 2001.
- [9] S. Mangold, S. Choi, P. May, O. Klein, G. Hiertz, and L. Stibor. IEEE 802.11e Wireless LAN for Quality of Service. In *European Wireless (EW'2002)*, volume 1, pages 32–39, Florence, Italy, February 2001.
- [10] Lamia Romdhani, Qiang Ni, and Thierry Turletti. Enhanced Service Differentiation for IEEE 802.11 Wireless Ad-Hoc Networks. *Wireless Communications and Network Conference 2003 (WCNC'03)*, March 2003.
- [11] J. Schiller. *Mobile Communications*. Addison-Wesley, 1st edition, 2000. ISBN 0201398362.
- [12] Ken Tang and Mario Gerla. Fair Sharing of MAC under TCP in Wireless Ad Hoc Networks. In *IEEE Multiaccess, Mobility and Teletraffic for Personal Communications (MMT'99)*, Venice, Italy, October 1999.
- [13] Nitin H. Vaidya, Paramvir Bahl, and Seema Gupta. Distributed Fair Scheduling in a Wireless LAN. In *ACM The Sixth Annual Conference on Mobile Computing and Networking (MobiCom'00)*, Boston, USA, August 2000.