# 4

# Performance Study of IEEE 802.11 DCF and IEEE 802.11e EDCA

Giuseppe Bianchi[a], Sunghyun Choi[b] and Ilenia Tinnirello[c]

## 4.1 Introduction

One of the key factors for the wide acceptance and deployment of IEEE 802.11 Wireless Local Area Networks (WLANs) is the simplicity and robustness of the Medium Access Control (MAC) protocol. Based on the well-known carrier sense paradigm, with an exponential backoff mechanism devised to minimize the probability of simultaneous transmission attempts by multiple stations, the protocol is able to work in presence of interference, which is very critical for networks operating in unlicensed spectrum. In fact, interfering sources are simply revealed by the carrier sense mechanism in terms of channel occupancy times, or by the acknowledgement mechanism in terms of collisions. However, the simplicity and the robustness have often been traded off with the efficiency of the access protocol, in terms of radio resources which are wasted or underutilized.

In this chapter, we provide a detailed analysis of the 802.11 distributed access protocol, by examining the protocol parameters which most critically affect the protocol efficiency. We quantify the protocol overheads due to control information (i.e., physical headers, frame headers, acknowledgement and other control frames) and to the distributed management of the channel grants (i.e., collisions and idle backoff slots). Then, we consider the distributed channel access extensions, defined in the recently-ratified 802.11e standard in order to support service differentiation among stations with different Quality-of-Service (QoS) requirements. Finally, we attempt to show how these parameters affect the resource repartitioning among the stations and how they can coexist with legacy DCF stations.

The IEEE 802.11 defines a basic service set (BSS) as the number of stations controlled by a single coordination function, where a coordination function is the 802.11 terminology for medium access control (MAC). There are two types of BSSs. One is the infrastructure BSS, and the other is the independent basic service set (IBSS). An infrastructure BSS is composed of a single access point (AP), (a bridge between an infrastructure, i.e., a wireline network typically, and the wireless link), and a number of

[a] *University of Roma Tor Vergata, Italy*
[b] *Seoul National University, Korea*
[c] *University of Palermo, Italy*

stations associated with the AP. Within an infrastructure BSS, a station communicate only with its AP, i.e., there is no direct transmission between two stations belonging to the same BSS. On the other hand, in an IBSS, there is no AP, and hence all the transmissions are between stations. We are primarily concerned with the IBSS in this chapter. The term "ad-hoc" is often used as an alternative term to refer to an IBSS in an IEEE 802.11 WLAN, and we use these terms equivalently throughout this chapter.

## 4.2 IEEE 802.11 MAC Protocol

The IEEE 802.11 legacy MAC [1] is based on logical functions, called the coordination functions, which determine when a station operating in a given 802.11 network is permitted to transmit and may be able to receive frames via the wireless medium. A data unit arriving from the higher layer to the MAC is referred to as a MAC Service Data Unit (MSDU), and the frame, which conveys the MSDU or its fragment along with the MAC header and Frame Check Sequence (FCS) based on CRC-32, is referred to as MAC Protocol Data Unit (MPDU). The MPDU is the frame that is transferred between stations from the MAC's perspective.

Two coordination functions are defined. The mandatory Distributed Coordination Function (DCF) allows distributed contention-based channel access based on carrier-sense multiple access with collision avoidance (CSMA/CA). The optional Point Coordination Function (PCF) provides centralized contention-free channel access based on a poll-and-response mechanism. Most of today's 802.11 devices operate only in the DCF mode. Accordingly, we limit ourselves to the DCF operation in this chapter.

The 802.11 DCF works with a single first-in-first-out (FIFO) transmission queue. The CSMA/CA mechanism is a distributed MAC protocol based on a local assessment of the channel status, i.e., whether the channel is busy (i.e., a station is transmitting a frame) or idle (i.e., no transmission). Basically, the CSMA/CA of the DCF works as follows.

When a frame arrives at the head of the transmission queue, if the channel is busy, the station waits until the medium becomes idle, and then defers for an extra time interval, called the DCF Interframe Space (DIFS). If the channel stays idle during the DIFS deference, the station then starts the backoff process by selecting a random backoff count. For each slot time interval, during which the medium stays idle, the random backoff counter is decremented. When the counter reaches zero, the frame is transmitted. On the other hand, when a frame arrives at the head of the queue, if the station is in either the DIFS deference or the random backoff process, the processes described above are applied again. That is, the frame is transmitted only when the random backoff has finished successfully. When a frame arrives at an empty queue and the medium has been idle longer than the DIFS time interval, the frame is transmitted immediately.

Each station maintains a contention window (CW), which is used to select the random backoff count. The backoff count is determined as a pseudo-random integer drawn from a uniform distribution over the interval [0,CW]. How to determine the CW value is further detailed as follows. If the channel becomes busy during a backoff process, the backoff is suspended. When the channel becomes idle again, and stays idle for an extra DIFS time interval, the backoff process resumes with the latest backoff counter value. The timing of DCF channel access is illustrated in Figure 4.1.

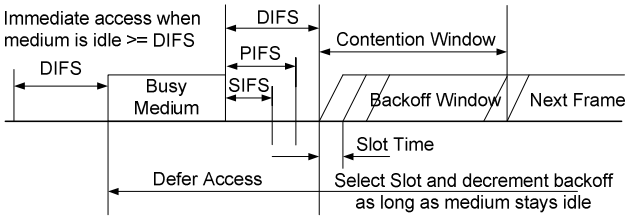


**Figure 4.1: IEEE 802.11 DCF channel access.**

For each successful reception of a frame, the receiving station immediately acknowledges the frame reception by sending an acknowledgement (ACK) frame. The ACK frame is transmitted after a short IFS (SIFS), which is shorter than the DIFS. Other stations resume the backoff process after the DIFS idle time. Thanks to the SIFS interval between the data and ACK frames, the ACK frame transmission is protected from other stations' contention. If an ACK frame is not received after the data transmission within the ACK Timeout[1], the frame is retransmitted after another random backoff.

The CW size is initially assigned $CW_{min}$, and increases when a transmission fails, i.e., the transmitted data frame has not been acknowledged. After an unsuccessful transmission attempt, another backoff is performed using a new CW value updated by

$$CW := 2(CW + 1) - 1, \tag{1}$$

with an upper bound of $CW_{max}$. This reduces the collision probability in case when there are multiple stations attempting to access the channel. After each successful transmission, the CW value is reset to $CW_{min}$, and the transmission-completing station performs the DIFS deference and a random backoff even if there is no other pending frame in the queue. This is often referred to as "post" backoff, as this backoff is done after, not before, a transmission. This post backoff ensures there is at least one backoff interval between two consecutive MPDU transmissions.

In the WLAN environment, there may be hidden stations. Two stations, which can transmit to and receive from a common station while they are out of range from each other, are known as hidden stations. Since the DCF operates based on carrier sensing, the existence of such hidden stations can severely degrade the network performance. To reduce the hidden station problem, 802.11 defines a Request-to-Send/Clear-to-Send (RTS/CTS) mechanism. If the transmitting station opts to use the RTS/CTS mechanism, then before transmitting a data frame, the station transmits a short RTS frame, followed by a CTS frame transmitted by the receiving station. The RTS and CTS frames include information of how long it takes to transmit the subsequent data frame and the corresponding ACK response. Thus, other stations hearing the transmitting station and hidden stations close to the receiving station will not start any transmissions; their timer called Network Allocation Vector (NAV) is set, and as long as the NAV value is non-zero, a station does not contend for the medium. Between two consecutive frames in the sequence of RTS, CTS, data, and ACK frames, a SIFS is used. Figure 4.2 shows the timing diagram involved with an RTS/CTS frame exchange. It should be noted that the RTS/CTS exchange can be very

[1] According to Annex C of [1], ACK Timeout is defined as SIFS + ACK transmission duration + SlotTime.

useful even if there is no hidden stations. For instance, when there are many contending stations, the bandwidth loss due to RTS collisions can be smaller than that due to longer frames [16].

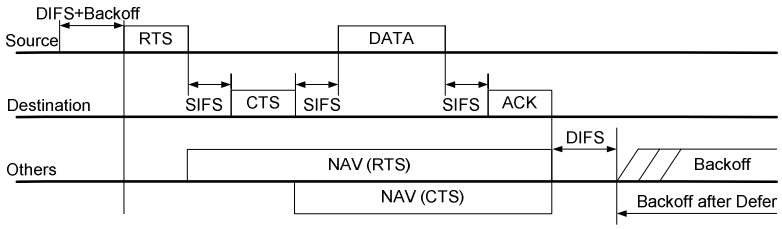


**Figure 4.2: RTS/CTS frame exchange.**

As explained thus far, the DCF normally waits for a DIFS interval before a backoff countdown. However, an EIFS interval shall be used instead for the contention immediately after an unsuccessful frame reception. Beginning any successful reception of a frame, the station starts using the DIFS instead of EIFS again. There are basically two different cases, which result in an unsuccessful frame reception: (1) the PHY has indicated the erroneous reception to the MAC, e.g., carrier lost; or (2) the error is detected by the MAC via an incorrect FCS value. The EIFS is defined to provide enough time for other stations to wait for the ACK frame of an incorrectly received frame before these stations start their frame transmission. Accordingly, the EIFS value is determined by the sum of one SIFS, one DIFS, and the time needed to transmit an ACK frame at the underlying PHY's lowest mandatory rate, which is 1 Mbps in the case of 802.11b PHY, i.e.,

$$EIFS = SIFS + ACK\_Tx\_Time_{@\,1(\mathrm{Mbps})} + DIFS \qquad (2)$$

Figure 4.3 illustrates that stations receiving the data frame incorrectly defer for an EIFS period before starting a backoff procedure.

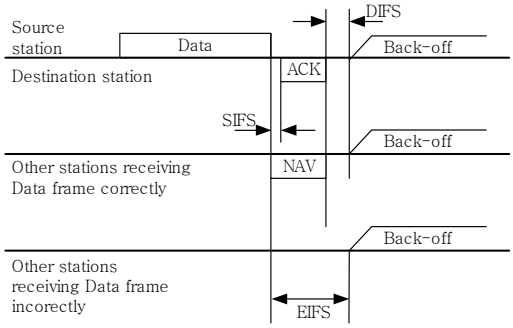


**Figure 4.3: DCF access operation, where the ACK frame is transmitted at 1 Mbps.**

All of the MAC parameters including SIFS, DIFS, Slot Time, $CW_{min}$, and $CW_{max}$ are dependent on the underlying PHY. Table 4.1 shows these values for the popular 802.11b PHY [6]. Irrespective of the PHY, DIFS is determined by SIFS+2•SlotTime, and another important IFS, called PCF IFS (PIFS), is determined by SIFS+SlotTime.

**Table 4.1: MAC parameters for the 802.11b PHY.**

| Parameters | SIFS (μsec) | DIFS (μsec) | Slot Time (μsec) | $CW_{min}$ | $CW_{max}$ |
|---|---|---|---|---|---|
| 802.11b PHY | 10 | 50 | 20 | 31 | 1023 |

### 4.2.1 DCF Overhead

It is very instructive to understand the protocol overhead introduced by the DCF operation. To this purpose, consider a scenario characterized by just a single transmitting station. For simplicity, neglect all the protocol overheads introduced by the upper layers (e.g., IP header, TCP/UDP header, etc.) as well as the interaction of the upper layers with the MAC operation (e.g., such as the TCP congestion control).

For a single transmitting station under the assumption that the frames are never corrupted by the channel noise, the maximum throughput can be immediately expressed as

$$S_{station} = \frac{E[payload]}{E[T_{Frame\_Tx}] + DIFS + SlotTime \cdot CW_{\min} / 2}, \quad (3)$$

where $SlotTime \cdot CW_{min}/2$ is the average time spent for the backoff between two consecutive data frame transmissions. Now, the time $T_{Frame\_Tx}$ spent to complete a frame transfer successfully depends on the considered handshake as well as the PHY employed. In the case of the basic access without an RTS/CTS exchange, this time duration is given by:

$$T_{Frame\_Tx} = T_{MPDU} + SIFS + T_{ACK}, \quad (4)$$

where $T_{MPDU}$ and $T_{ACK}$ represent the transmission times of an MPDU (or a data frame) and an ACK, respectively. These are dependent on the employed transmission rate as well as the frame size. Note that PHYs of the 802.11 provide a set of transmission rates. For example, the 802.11b PHY [6] provides four different transmission rates, namely, 1, 2, 5.5, and 11 Mbps. Which transmission rate to use for a particular frame transmission is not defined in the standard, and is left implementation-dependent. Some rate adaptation algorithms can be found in [12 - 15].

Now, for the transmission rate $R_{MPDU\_Tx}$, the transmission time of a MPDU conveying a payload of $L$ bytes is determined by:

$$T_{MPDU} = T_{PLCP} + 8 \cdot (28 + L) / R_{MPDU\_Tx}, \quad (5)$$

where $T_{PLCP}$ is the overhead due to the PHY operation, e.g., preamble and header, and is given by 192 μsec for the case when the 802.11b PHY employs the long-preamble option. The number 28 in Eq. (5) represents the overhead of the MAC header plus the FCS field in number of bytes. Similarly, the transmission time of an ACK frame, which is 14 bytes long, for the transmission rate $R_{ACK\_Tx}$ can be represented by:

$$T_{ACK} = T_{PLCP} + 8 \cdot 14 / R_{ACK\_Tx} \tag{6}$$

With RTS/CTS exchange, the time to complete a frame transfer is given by:

$$T_{Frame\_Tx} = T_{RTS} + SIFS + T_{CTS} + SIFS + T_{MPDU} + SIFS + T_{ACK}, \tag{7}$$

where the RTS (of 20 bytes) and CTS (of 14 bytes) transmission times are given, respectively, by

$$\begin{aligned} T_{RTS} &= T_{PLCP} + 8 \cdot 20 / R_{RTS\_Tx} \\ T_{CTS} &= T_{PLCP} + 8 \cdot 14 / R_{CTS\_Tx} \end{aligned}. \tag{8}$$

Figure 4.4 presents the analysis of different DCF overheads for two different 802.11b 802.11b transmission rates for the MPDU when the payload is fixed at 1500 bytes. Irrespective of the MPDU transmission rate, 1 Mbps (minimum rate) was assumed for the transmission of the ACK, RTS, and CTS frames. We observe that the protocol overheads due to the backoff, RTS, CTS, and ACK are relatively large when the MPDU transmission rate is high since the time corresponding to the payload transmission is relatively short for a high transmission rate. We can also easily envision that the overheads will be relatively large as the payload size is reduced since the overheads are relatively fixed in time.

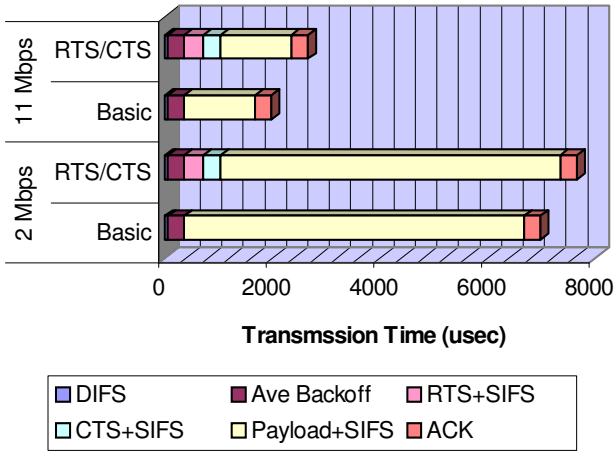


**Figure 4.4: Analysis of different DCF overhead for 1500 byte-long payload.**

This is more clearly presented in Figure 4.5, which illustrates the normalized throughput, defined by the throughput divided by the MPDU transmission rate. We first observe that the normalized throughputs increase as the payload size increases. Second, we observe that the normalized throughputs are larger for a lower transmission rate due to the relatively smaller protocol overhead as observed in Figure 4.4. Finally, in the considered simple situation, i.e., a single transmitting station, the RTS/CTS exchange obviously results in a lower throughput performance.

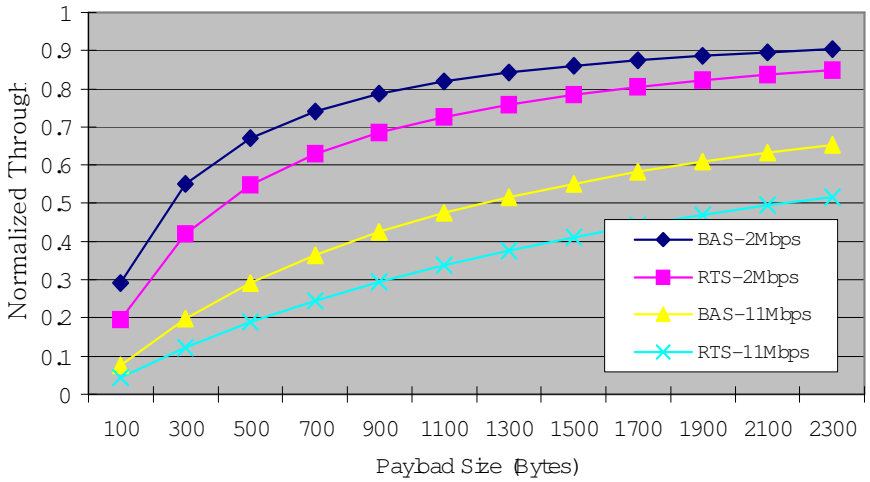


**Figure 4.5: Normalized throughput versus payload size (bytes) for different MPDU transmission rates.**

## 4.3 Performance Evaluation of the IEEE 802.11 DCF

Under suitable simplifying assumptions, it has been proven in [16] that the accurate performance evaluation of the DCF employed in the IEEE 802.11 legacy MAC can be carried out using elementary analytical techniques. The availability of such techniques allows us to easily obtain quantitative insights on the effectiveness of the DCF mechanism and the related parameter settings. The goal of this section is to provide the reader with a comprehensive overview of the modelling techniques suitable for evaluating the DCF performance. The reader interested in additional modelling details and extensions may refer to [16 - 25].

### 4.3.1 The Concept of Saturation Throughput

In Section 4.2.1, we have derived the maximum throughput that a single transmitting station can achieve, and we have quantified the DCF protocol overhead. From a practical point of view, this implies that all the traffic arriving at a long-term rate lower than the maximum throughput value will be delivered to the destination. Conversely, as long as the traffic arrival rate persistently grows above the maximum throughput threshold, the transmission buffer will build up until saturation, and the carried load will remain bounded to the maximum throughput value.

In the case of several competing stations, the situation is different. It is well known[2] that several random access schemes exhibit an unstable behaviour. In particular, as the offered load increases, the throughput grows up to a maximum value, referred to as

[2] See for example the well-known textbook by Dimitri Bertsekas and Robert Gallager, Data Networks, 2nd edition, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1992 – this book provides a comprehensive and thorough discussion of the instability problems arising in random access protocols.

"Maximum Throughput." However, further increases of the offered load lead to a significant decrease in the system throughput eventually (which typically converges to zero in the case of infinite users).

Mild forms of instability arise also in the case of DCF's Binary Exponential Backoff operation, and for a scenario characterized by a finite number of competing stations. This can be illustrated by means of a simple simulation experiment. The plots shown in Figure 4.6 (taken from [16] – the simulation details and the throughput/load scales are not essential for the following discussion, and hence are omitted) have been obtained considering a finite number of stations. Each station has been loaded with a variable amount of traffic, linearly increasing with the simulation time (straight line). Fixed-size MPDUs have been considered.

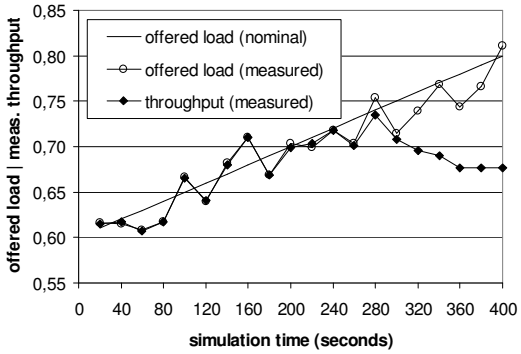


**Figure 4.6: Throughput versus offered load.**

The figure reports two additional plots: the offered load and the throughput, measured in time intervals lasting 20 seconds. Clearly, due to statistical fluctuations of the packet arrival process (Poisson in the specific case), the measured offered load in general differs from the nominal load. However, in normal (stable) conditions, the entire offered load will be delivered to the destination, and hence the measured throughput will be equal to the measured offered load. From the figure, we see that this happens only in the first part of the simulation run, specifically the first 260 seconds of the simulation time. After this time, the throughput measured in each 20-second time interval becomes smaller than the measured offered load (and excessive frames accumulate in the transmission buffer).

From the figure, we also see that the measured throughput appears to asymptotically converge to a constant value (0.68 in this specific example), regardless of the offered load. We define "saturation throughput" as the limit reached by the system throughput as the offered load increases.

The importance of the saturation throughput concept stays in the fact that it represents the maximum load that the system can carry in *stable conditions*, i.e., in practical operation. In fact, refer again to Figure 4.6, and consider, for an example, the load scenario encountered at simulation time 280s. Here, the nominal offered load is approximately 0.74, while the load measured in the 20-second time interval is approximately 0.75. If we now freeze the offered load to a constant value, i.e., 0.74, and we take measurements on a longer time interval, we would see that the longer the measurement time, the lower the measured throughput (ultimately, for an infinite measurement period, the measured throughput will result in a value equal to the saturation value).

Actually, the same situation will occur for *any* load greater than the saturation throughput bound, provided that a sufficiently long measurement time is considered. This results in the practical impossibility to maintain a sustained operation of the random access scheme at any load greater than the saturation value[3].

### 4.3.2 Maximum Saturation Throughput

Having defined the concept of saturation throughput, we are now ready to describe an analytical approach which allows us to derive this performance figure. In this section, we derive performance bounds on the throughput achievable by DCF [16, 26, 27]. Consider an 802.11 ad-hoc network scenario in which a finite and fixed number *N* of stations contend for the channel access. Assume that each station is in a saturation condition, meaning that its buffer is always non-empty and, at any time, a frame is always available for transmission. Moreover, assume that a frame transmission is never corrupted (either by noise or by interference due to hidden terminals), and that a transmission fails only when a collision with another frame occurs on the channel (this assumption of ideal channel conditions can be removed, as shown in Section 4.3.4 – see also [18, 21]).

The first important observation is that the 802.11 DCF rules allow us to introduce a discrete-time integer time-scale, which is the key to enable the model described below. As described in Section 4.2, a station can transmit only when it senses the channel idle for a DIFS. Moreover, after this time, it can schedule transmission only in discrete slot-time intervals, whose size is hereafter indicated as $\sigma$. Moreover, let us focus on the events occurring on the channel:

- When only one station has a scheduled transmission in a slot-time, neglecting wireless channel impairments, the transmission will be successful; all the other stations will freeze their backoff counters, until a DIFS time elapses after the end of the ACK.
- When two or more stations schedule transmission into a slot-time, a collision will occur, and the channel will be available for access only a DIFS or an EIFS[4] after the end of the longest transmitted frame.
- Finally, if no stations transmit in the given slot-time, the next transmission opportunity will be the following slot-time.

[3] Though traditional performance evaluation models for random access schemes frequently derive a theoretical maximum throughput value, this value results meaningless from a practical point of view. Actually, this value is not even measurable from a simulation experiment. For example, in the case of Figure 4.6, we obtain a maximum measured throughput equal to about 0.74, but this measurement is only a rough estimate, as it is affected by the unreliability of a fairly short measurement time. If we rerun the same simulation experiment using measurement times longer than 20 seconds (and for consistency, we consider a slower increase of the nominal offered load), the measurement will be more reliable, but the maximum value will be lower.

[4] It is not trivial to determine whether a specific listening station will use a DIFS or an EIFS after a frame collision. If the listening station is able to synchronize with one of the colliding frames, and thus initiate a receiving process, an EIFS will be used (see the reason in Section 4.2). On the other hand, if this is not the case (e.g., comparable received power level for the two colliding frame preambles), the listening station will just see a busy channel: without initializing the reception of any frame, a DIFS will be used. All the simulation (as well as analytical) models we are aware of, for simplicify do not enter into this technical issue, but consistently use either DIFS or EIFS for all the collisions (for example, the 802.11 ns-2 implementation uses EIFS, and this justifies some different results with respect to other models that use DIFS after a collision).

Hence, a discrete and integer time scale can be defined. Note that this discrete time scale does not directly relates to the system time, being a "slot" on the channel either an empty slot (in which case, the slot will last exactly one slot-time $\sigma$), or a busy slot, in which case the slot duration will depend on the events occurring (a transmission or a collision). In what follows, unless ambiguity occurs, with the term slot time, we will refer to either the (constant) value $\sigma$, representing the system slot-time, or the (variable) time interval representing the model slot time.

To determine the maximum throughput achievable in an 802.11 ad-hoc network composed of $N$ saturated stations, let us now assume that each station randomly and independently accesses a slot time with probability $\tau$. This equals to assume that every station follows a $p$-persistent backoff strategy, where the probability to access a random slot is constant and set to the value $\tau$. In this assumption, the probability $P_{idle}$ that no station accesses a given slot is readily given by:

$$P_{idle} = (1-\tau)^N \tag{9}$$

Similarly, the probability $P_{success}$ that just one station accesses a given slot is expressed as:

$$P_{success} = N\tau(1-\tau)^{N-1} \tag{10}$$

Let us now define $T_s$ to be the duration of a period in which no other stations can access the channel because a successful transmission is being occurring. This period not only includes the MPDU transmission time as well as the relevant ACK transmission time, but also includes a DIFS after the end of the ACK transmission, since in this period of time no other station can access the channel. Similarly, let $T_c$ be the duration of a period in which other stations cannot access the channel because a collision is occurring. In what follows, we will refer to these two values as transmission slot and collision slot durations, respectively, and these values will be expressed in Section 4.3.2.1. Since a system slot-time $\sigma$ elapses during an idle slot, we can derive the average slot duration by weighting in probability the three values $T_s$, $T_c$ and $\sigma$:

$$E[slot] = P_{idle}\sigma + P_{success}T_s + (1 - P_{idle} - P_{success})T_c \tag{11}$$

We are finally ready to define the system throughput $S$ as the average amount of information transmitted into a slot. Given that $E[P]$ is the average MPDU payload size,

$$S = \frac{P_{success}E[P]}{E[slot]} = \frac{P_{success}E[P]}{P_{idle}\sigma + P_{success}T_s + (1 - P_{idle} - P_{success})T_c} = \\ = \frac{E[P]}{T_s + \sigma\frac{P_{idle} + T_c^*(1 - P_{idle} - P_{success})}{P_{success}}}, \tag{12}$$

where $T_c^* = T_c/\sigma$ is the average collision time measured in slot-time units. Now, $E[P]$, $\sigma$, and $T_s$ are constant values. Hence, the throughput above is maximized as long as we minimize the expression:

$$\frac{P_{idle} + T_c^*\left(1 - P_{idle} - P_{success}\right)}{P_{success}} = \left(1 - T_c^*\right)\frac{(1-\tau)}{N\tau} + \frac{T_c^*}{N\tau(1-\tau)^{N-1}} - T_c^* \tag{13}$$

If we approximate the average value $T_c^*$ to be a constant value[5], independent on $\tau$, we finally conclude that the optimal value $\tau_{\max}$ that maximizes the system throughput is given by the solution of the equality:

$$(1-\tau_{\max})^N - T_c^*\left\{N\tau_{\max} - \left(1 - (1-\tau_{\max})^N\right)\right\} = 0 \tag{14}$$

Under the condition $\tau_{\max} << 1$, the approximation

$$(1-\tau_{\max})^N \approx 1 - N\tau_{\max} + \frac{N(N-1)}{2}\tau_{\max}^2, \tag{15}$$

holds, and hence an explicit expression for the equality in Eq. (14) can be found:

$$\tau_{\max} = \frac{\sqrt{1 + 2\left(T_c^* - 1\right)\frac{(N-1)}{N}} - 1}{(N-1)\left(T_c^* - 1\right)} \approx \frac{1}{N\sqrt{T_c^*/2}}, \tag{16}$$

where the last approximation holds for large values of $N$ and $T_c^*$. An interesting alternative way to express Eq. (16) is to determine the optimal value of the contention window, $CW_{\text{opt}}$, which allows us to maximize the throughput. Specifically, consider a station which, instead of using exponential backoff rules, always extracts the backoff counter from the uniform range $[0, CW_{opt}]$. Under the assumption that the backoff counter is decremented at each slot (see a related discussion in Section 4.3.3.1), a transmission occurs every $(1+CW_{opt}/2)$ slots. Hence, the probability that a station transmits in a randomly chosen slot can be related to the Contention Window as:

$$\tau = \frac{1}{1 + CW_{opt}/2} \tag{17}$$

Substituting Eq. (17) into the $\tau_{\max}$ expression given in Eq. (16), we conclude that $CW_{opt}$ is, in the first approximation, proportional to the number $N$ of competing stations, i.e.,

[5] This is strictly true only if the frames have fixed-size payload. In fact, with variable payload size, the length of a collision interval is bounded by the length of the longest packet involved in the collision. This gets longer as long as the number of packets simultaneously colliding gets greater. As the probability that a given number of packets are involved in a collision depends on $\tau$, the value $T_c^*$ is actually a function of $\tau$.

$$CW_{opt} \approx 2N\sqrt{T_c^*/2} - 2 \approx N\sqrt{2T_c^*} \tag{18}$$

#### 4.3.2.1 Performance Bounds for 802.11b DCF

To compute the performance bounds for the DCF, we need to quantify[6] the parameters $\sigma$, $T_s$, and $T_c$. Similar to what have been done in Section 4.2.1, we refer to the 802.11b case; accordingly, the slot-time $\sigma$ is set to 20 μsec. $T_s$ is given by the time to transmit a frame ($T_{Frame\_Tx}$, computed in Section 4.2.1) plus a DIFS interval. For the case of the basic access:

$$T_s = T_{MPDU} + SIFS + T_{ACK} + DIFS\,, \tag{19}$$

where $T_{MPDU}$ and $T_{ACK}$ have been computed in Eqs. (5) and (6), respectively (there, instead of $E[P]$, we have referred to the MPDU payload size as $L$). Similarly, for the case of RTS/CTS exchange,

$$T_s = T_{RTS} + SIFS + T_{CTS} + SIFS + T_{MPDU} + SIFS + T_{ACK} + DIFS \tag{20}$$

Refer to Eq. (8) for $T_{RTS}$ and $T_{CTS}$ expressions. The computation of $T_c$ differs depending on whether we assume that a DIFS or an EIFS is used after a collision (see detailed discussion in footnote 4). Though it is perhaps more realistic, for a small scale single hop ad-hoc network, to assume destructive collisions, and thus that a DIFS elapses after a collision, in what follows, we will use an EIFS after a collision since such a setting requires more discussion on the station timing (in other words, the extension of what follows to the DIFS case is just a simplification). If all the frames have the same size, for the basic access case,

$$T_c = T_{MPDU} + EIFS \tag{21}$$

For the general case of frames having different sizes, then the duration of a collision depends also on the number of colliding frames, and thus on the transmission probability $\tau$ (refer to Eq. (15) in [16] for details). Instead, in the RTS/CTS case, collisions occur only for the RTS frames. Therefore, the duration of a collision is constant, and is given by

$$T_c = T_{RTS} + EIFS \tag{22}$$

Figure 4.7 presents the saturation throughput performance for 10 stations and 802.11b parameters (see Section 4.2.1). The case of 2 and 11 Mbps data rates are plotted, for both basic and RTS/CTS access cases. The rate for control frames (i.e., ACK, RTS, and CTS) is set to the minimum rate (i.e., 1 Mbps). The circle represents the maximum throughput computed by means of the approximate value $\tau$ given in Eq. (16). We first observe that the approximation in Eq. (16) is accurate. We find that, in the basic access case, the maximum

[6] The discussion tackled in this section will be further extended in Section 4.3.3.1, where additional considerations on the implication of the backoff countdown rules on $T_s$ and $T_c$ settings will be provided.

throughput performance is more sensitive to the value $\tau$ compared with the RTS/CTS case. This implies that the RTS/CTS access mechanism is less sensitive to backoff parameters (as we will show in Section 4.3.3, DCF operates to a value $\tau$ which actually depends on the backoff settings, i.e., $CW_{min}$, $CW_{max}$, etc). Moreover, Figure 4.7 shows that, in the case of low data rates (2 Mbps), the maximum throughput achievable by basic access and RTS/CTS exchange are very close. This is not true any more when higher data rates (11 Mbps) are considered, and the overhead due to the RTS/CTS exchange becomes a limiting factor in terms of achievable throughput.

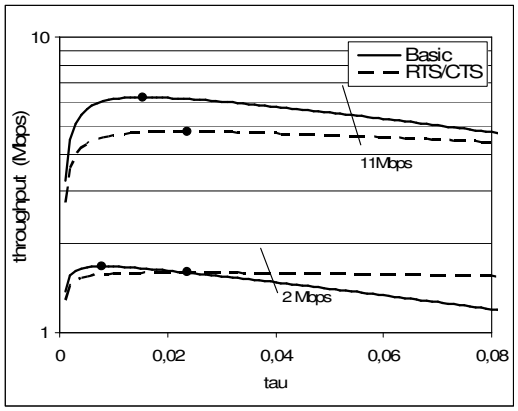


**Figure 4.7: The maximum throughput with the number of stations, *N*=10.**

Finally, let us evaluate the effect of the number of stations on the maximum achievable throughput. For convenience of notation, let

$$K = \sqrt{T_c^* / 2} \tag{23}$$

By using the approximation for the optimal τ given by Eq. (16), recalling that $T_c^* = T_c/\sigma$, and taking the limit for $N \to \infty$,

$$\begin{aligned} \lim_{N\to\infty} S_{\max} &= \lim_{N\to\infty} \frac{E[P]}{T_s + \sigma \dfrac{P_{idle} + T_c^*\left(1 - P_{idle} - P_{success}\right)}{P_{success}}} = \\ &= \lim_{N\to\infty} \frac{E[P]}{T_s + \sigma \dfrac{(1-\tau_{\max})}{N\tau_{\max}} - T_c + T_c \dfrac{\left(1-(1-\tau_{\max})^N\right)}{N\tau_{\max}(1-\tau_{\max})^{N-1}}} = \\ &= \frac{E[P]}{T_s + \sigma K - T_c(1 + K - Ke^{1/K})} \end{aligned} \tag{24}$$

We thus conclude that, even for a large number of stations, the maximum throughput tends to a constant finite value, which is a function of only the average transmission and

collision times, $T_s$ and $T_c$, and the slot size. It is interesting to note, via direct computation, that such an asymptotic maximum throughput is very close to that achievable in the case $N$=10 given in Figure 4.7. In fact, through Eq. (24), we derive that, in the 2 Mbps data rate case, the asymptotic throughput is 1.669 and 1.596 Mbps for the basic and RTS/CTS cases, respectively, while it results in 6.210 (basic) and 4.763 (RTS/CTS) Mbps for the 11 Mbps data rate scenario, respectively.

### 4.3.3 Saturation Throughput Analysis

Having derived the capacity limits of the IEEE 802.11 DCF, we now carry out an analysis devised to understand how far from its performance limits DCF operates. Such an analysis appears more complex: since each station accesses the channel according to Binary Exponential Backoff rules, the space state required to thoroughly model each individual station (e.g., the number of retransmission suffered by each station, and the backoff counter value) rapidly diverges, even in the presence of a small number of competing stations.

However, let us focus on a specific station, hereafter referred to as "tagged" station. This station will access the channel according to the Binary Exponential Backoff mechanism specified for DCF, and specifically, as described in Section 4.2, it will double the range in which the Contention Window is chosen every time a collision is encountered. Hence, the tagged station will access the channel with a "frequency" (measured in terms of number of accesses per channel slot) which depends on the number of retransmissions already suffered by the considered frame: a high frequency when the CW value is small, a small frequency conversely. In turns, each of the remaining competing stations will be characterized by complex exponential backoff rules and, thus, very different Contention Window values CW, depending on the specific history of each access attempt (e.g., the number of retransmissions suffered by the actual head-of-line MPDU). However, in stationary conditions, we argue that it is reasonable to consider their "aggregate" contribution as being, statistically speaking, invariant, and specifically to consider their effect as the result of individual stations accessing the channel via a suitable (i.e., to be determined), but constant, permission probability.

Such an intuitive statement can be formally reworded by means of the two following key assumptions:

1. Regardless of the history of the head-of-line frame in terms of the number of retransmissions and accumulated backoff stage, we assume that each frame transmission suffers from a constant and independent collision probability;
2. Being $p$ such a collision probability, and being $N$ the number of competing stations, we assume that $p$ is computed as the contribute of $N$-1 remaining stations, each independently accessing a channel slot with a constant permission probability $\tau$.

As shown in what follows, these assumptions enable a very simple, though accurate, analytical modelling of the DCF.

For the sake of generality, it is useful to develop the model considering backoff rules more general than the exponential backoff specified in the DCF standard. To this purpose, let us define with the term "Backoff Stage" the number of retransmissions suffered by a Head-Of-Line (HOL) frame. A station in backoff stage 0, i.e., willing to transmit a new

MPDU, will select[7] an integer random backoff value drawn from a general probability distribution $B_0$. If the transmitted frame collides, we say that the station enters backoff stage 1. The next backoff value will be drawn from a second probability distribution $B_1$, and so forth. In general, a station entering backoff stage $i$ will extract a backoff value from a distribution $B_i$.

In the particular case of the DCF Binary Exponential Backoff, $B_0$ is a uniform distribution in the range $[0,CW_{\min}]$, $B_1$ is a uniform distribution in the range $[0, 2(CW_{\min}+1)-1]$ and, in general, $B_i$ is a uniform distribution in the range $[0,CW_i]$, being $CW_i=2^i(CW_{\min}+1)-1$. In addition, the IEEE 802.11 DCF specifies

1. the maximum Contention Window value $CW_{\max}=2^m(CW_{\min}+1)-1$, being $m$ a parameter depending on the physical layer considered, and
2. a finite number of retries $R$, meaning that a frame whose first transmission has failed, will be retransmitted for at most $R$ times, and then it will be dropped from the transmission queue.

In what follows, we will show that the performance do not depend on the probability distributions $B_i$, but only on their mean values $\beta_i = E[B_i]$. Moreover, we will show that the performance depend on the retry limit $R$, being $R$ eventually infinite in the analytical model.

Let us denote with $(TX)$ the event that a station is transmitting a frame into a time slot, and denote with $(s=i)$ the event that the station is found in backoff stage $i$.

We are ultimately interested in the unconditional probability $\tau = P\{TX\}$ that the station transmits in a randomly chosen slot. Thanks to Bayes' Theorem, for $i \in (0, \ldots, R)$,

$$P\{s=i \mid TX\} = \frac{P\{TX \mid s=i\}P\{s=i\}}{P\{TX\}} \tag{25}$$

which in turn can be rewritten as:

$$P\{TX\}\frac{P\{s=i \mid TX\}}{P\{TX \mid s=i\}} = P\{s=i\} \tag{26}$$

Since this equality holds for all $i \in (0, \ldots, R)$, it also holds for the summation:

$$\sum_{i=0}^{R} P\{TX\}\frac{P\{s=i \mid TX\}}{P\{TX \mid s=i\}} = \sum_{i=0}^{R} P\{s=i\} \tag{27}$$

However, the rightmost term in the equation is a probability distribution (namely, the probability that a station is in backoff stage $i$). Hence, the sum over all $i \in (0, \ldots, R)$ equals 1. We can thus derive an expression for $\tau$.

[7] Saturation conditions imply that a packet in backoff stage 0 immediately follows a previously transmitted one. Hence, consistently with the DCF specifications (see Clause 9.1.1 of the standard), a random backoff interval shall be always selected also for the first packet transmission attempt.

$$\tau = P\{TX\} = \frac{1}{\sum_{i=0}^{R} \frac{P\{s=i \mid TX\}}{P\{TX \mid s=i\}}} \tag{28}$$

The value $\tau$ is thus known, as long as we find an expression for $P\{s=i|TX\}$ and $P\{TX|s=i\}$. Let us first focus on the conditional probability $P\{s=i|TX\}$ that a transmitting station is found in backoff stage $i$. Since, for $i>0$, this probability is given by the probability that the station, in the previous transmission event, was found in stage $i$-1 and that the transmission failed (by assumption, this occurs with constant probability $p$), it follows that $P\{s=i|TX\}$ is a (truncated, in the case of finite value $R$) geometric distribution[8], i.e.:

$$P\{s=i \mid TX\} = \frac{(1-p)p^i}{1-p^{R+1}} \qquad i \in (0,...,R) \tag{29}$$

Let us now find an explicit expression for $P\{TX|s=i\}$. This represents the transmission probability of a station in backoff stage $i$, or, in other words, the frequency of transmission (the number of transmission slots per channel slot) for a station assumed to always remain in backoff stage $i$. Under very general conditions[9], this probability can be computed dividing the average number of slots spent in the transmission state while in stage $i$ (owing to the time scale adopted, exactly 1 slot), with the average number of total slots spent by the station in stage $i$ (i.e. the average number of backoff slots, plus the single transmission slot). According to the notation given above:

$$P\{TX \mid s=i\} = \frac{1}{1+E[B_i]} = \frac{1}{1+\beta_i} \qquad i \in (0,...,R) \tag{30}$$

In the special case of DCF, a station entering backoff stage $i$ uniformly selects a backoff value in the range $[0,CW_i]$. Following [16], it is convenient to adopt the notation $W_i=CW_i+1$. Hence,

---

[8] A more formal way to derive Eq. (29) is to envision $P\{s=i|TX\}$ as the steady-state probability distribution of a discrete-time mono-dimensional Markov Chain describing the backoff stage evolution. One time step in this chain represents a backoff stage transition, driven by the success/failure of the packet transmission. At stage $0 \le i < R$, the chain will evolve in the next time step in stage $i+1$ with probability $p$, and will return (or stay) in stage 0 with probability $1-p$; at stage $R$ the chain will in any case return to stage 0 with probability 1. This interpretation allows us to simply extend the described analysis to more general backoff processes with memory, i.e., whose backoff evolution is regulated by a Markov chain. It suffices to substitute Eq. (29) with the steady-state distribution of the considered Markov Chain. Literature work reports a few proposals of backoff models with memory (for example, the slow CW decrease approach considered in [25]).

[9] Since we are conditioning on the backoff stage $i$, we can envision the event of transmitting into a slot as the recurrence of transmission events separated by the time spent while in backoff stage $i$, assumed independent among transmission events. Hence, this computation can be interpreted as an application of the Long-Run Renewal rate theorem (see, e.g., William Feller, An introduction to probability Theory and Its Applications, Vol. II, Wiley, Cap. XI - pp. 368-380) and is shown to depend only on the average time spent while in the backoff stage $i$, and not on its distribution. As a side comment, it has been sometimes proposed to draw backoff counters from distributions different from the usual uniform one. As it should be clear now, such a generalization appears of no practical significance, as performance depend only upon the mean value, and thus there's no reason in using backoff distributions more complex than the uniform one.

$$P\{TX \mid s=i\} = \frac{1}{1+E[uniform(0,CW_i)]} = \frac{1}{1+\frac{W_i-1}{2}} = \frac{2}{W_i+1} \tag{31}$$

By substituting Eqs. (29) and (30) into Eq. (28) we can finally derive an explicit expression for $\tau$:

$$\tau = \frac{1}{\sum_{i=0}^{R} \frac{(1-p)p^i}{1-p^{R+1}}(1+\beta_i)} = \frac{1}{1+\frac{(1-p)}{1-p^{R+1}}\sum_{i=0}^{R} p^i \beta_i}, \tag{32}$$

where we have made use of the fact that $P\{s=i|TX\}$ is a distribution which, in the considered range $(0,\ldots,R)$, sums to 1.

This expression depends on the values $R$ (retry limit), and the sequence of values $\beta_i$ (the mean per-stage backoff values), which are specified by the employed backoff model. Moreover, it depends on the conditional collision probability $p$ which is still unknown. To find the value of $p$ it is sufficient to note that the probability $p$ that a transmitted frame encounters a collision, is the probability that, in a time slot, at least one of the $N$-1 remaining stations transmits. The fundamental independence assumption #1 given at the beginning of this section implies that each transmission "sees" the system in the same state, i.e., in steady state. At steady state, according to assumption #2, each remaining station transmits a frame with constant permission probability $\tau$. This yields:

$$p = 1-(1-\tau)^{N-1} \tag{33}$$

Eqs. (32) and (32) represent a non linear system in the two unknowns $\tau$ and $p$, which can be solved using numerical techniques.

The above analysis was carried out for general backoff models. By properly choosing the sequence $\beta_i$ and the value $R$, it can be immediately adapted to more specific backoff model, e.g., the binary exponential backoff adopted in DCF. For example, by setting:

$$\begin{aligned} &R = \infty \\ &W = CW_{\min} + 1 \\ &m = \log_2\left(\frac{CW_{\max}+1}{CW_{\min}+1}\right) \\ &\beta_i = \begin{cases} \frac{2^i W - 1}{2} & 0 \le i \le m \\ \frac{2^m W - 1}{2} & i \ge m \end{cases} \end{aligned} \tag{34}$$

We model a Binary Exponential Backoff scheme with no retry limit as an upper bound on the Contention Window (summarized by the value *m*). Eq. (32) becomes [16]:

$$\tau = \frac{2}{1+(1-p)\left\{\sum_{i=0}^{m-1} p^i 2^i W + \sum_{i=m}^{\infty} p^i 2^m W\right\}} = \\ = \frac{2(1-2p)}{(1-2p)(W+1)+pW(1-(2p)^m)} \tag{35}$$

The same DCF Binary Exponential Backoff model, but with a finite retry limit *R* (for simplicity of computation lower than or equal to the parameter *m* – for the general case refer to [17]) yields:

$$\tau = \frac{2(1-2p)(1-p^{R+1})}{(1-2p)(1-p^{R+1})+W(1-p)(1-(2p)^{R+1})} \tag{36}$$

#### 4.3.3.1 Throughput Performance

Once the value $\tau$ is known, the throughput can be computed via Eq. (12), here reported for the convenience of the reader:

$$S = \frac{P_{success} E[P]}{P_{idle}\sigma + P_{success} T_s + (1-P_{idle}-P_{success})T_c} \tag{37}$$

where

$$P_{idle} = (1-\tau)^N \\ P_{success} = N\tau(1-\tau)^{N-1} \tag{38}$$

We recall that the denominator in Eq. (37) represents the average slot size *E*[*slot*]. If *E*[*P*], instead of bits, is expressed in the same time unit of the parameters at the denominator (e.g., seconds), then Eq. (37) gives the "normalized" saturation throughput, defined as the fraction of channel time used to send successful payload information.

The fundamental difference with respect to the treatment suggested in Section 4.3.2 is that, now, $\tau$ is no more a variable (i.e. a generic permission probability), but it is a numerical value function of the considered backoff model parameters, namely, the retry limit *R* and the sequence of mean per-stage backoff values $\beta_i$, for all *i* in (0,…,*R*).

The values $T_s$ and $T_c$ have been computed in Section 4.3.2.1. However, some further remarks are needed when the analysis is applied to the DCF. The IEEE 802.11 standard discusses, in Clause 9.2.5.2, how the backoff counter is decrement. Here, it specifies that, if the medium is determined to be busy at any time during a backoff slot, then the backoff procedure is suspended (meaning that the backoff timer shall not decrement for that slot). Hence, assume that a station has a backoff counter equal to a value *b* at the beginning of a

slot-time. If the current slot-time is idle, at the end of the slot-time the backoff counter is duly decremented, and the station will start the next slot-time with backoff value *b*-1. Conversely, if the current slot-time results to be busy (because another station starts transmitting in the considered slot), the station freezes the backoff counter to the value *b*. This implies that the station starts the slot immediately following a busy one with the same backoff value *b*. In other words, the backoff counter is decremented only during idle slots.

This operation has some not immediately evident implications on the modelling framework described until now. Figure 4.8 illustrates what happens when two stations access the channel with different backoff values. In the example, at slot t, stations A and B start with a backoff counter equal to 2 and 3, respectively. Hence, we might expect them to transmit in consecutive slots, namely slot t+2 and slot t+3.

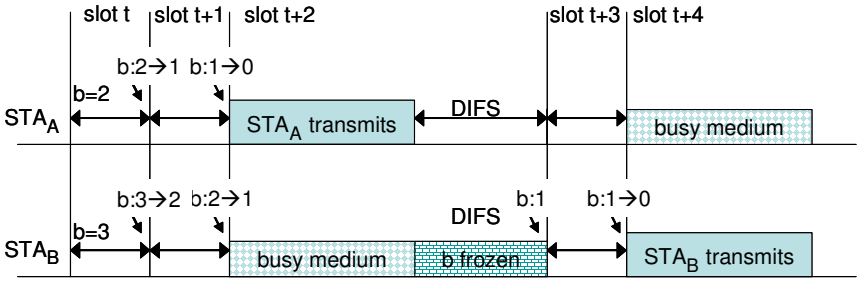


**Figure 4.8: The slot immediately following a transmission can be accessed only by the transmitting station.**

As expected, station A decrements the backoff counter to 0 at the end of slot t+1, thus transmiting a frame in slot t+2. It can then schedule the transmission for the next frame. With probability $1/(CW_{\min}+1)$, it will extract 0 as the new backoff counter, and hence it will immediately transmit the next frame in the first slot available, a DIFS after the end of the transmission (slot t+3, in the figure). Let us now focus on station B. In slot t+2, it will see station A's transmission on the channel and will freeze the backoff counter. It will thus start slot t+3 with a backoff counter value equal to 1, and (assuming slot t+3 to be empty) it will ultimately transmit only in slot t+4. We conclude that *a slot immediately following a successful transmission cannot be used for transmissions by any other station, except the transmitting one*. Hence, in ideal channel conditions, such a transmission is granted to be successful as no collision may occur.

The described effect can be accounted in the model by redefining the notion of successful transmission slot, and specifically by including either i) the extra slot-time at the end of a transmission, as well as ii) the possible extra frames transmitted in the "reserved" slot. In formulae:

$$\overline{T_s} = T_s + \sum_{k=1}^{\infty}\left(\frac{1}{CW_{\min}+1}\right)^k T_s + \sigma = T_s \frac{CW_{\min}+1}{CW_{\min}} + \sigma, \qquad (39)$$

where $T_s$ is the successful slot size given in Eqs. (19) and (20) for the Basic and RTS/CTS cases, respectively, which accounts for a single frame transmission plus a DIFS.

Consistently, in the throughput computation, the amount of information transmitted into a successful slot shall also include the MPDU payload due to extra frames transmitted in such "reserved" slots:

$$\overline{E[P]} = E[P]\frac{CW_{\min}+1}{CW_{\min}} \tag{40}$$

Since a successful "transmission slot" now includes an extra idle slot, a further model detail is that, in the slot immediately after such a transmission slot, the backoff counter will be found in the range (0, $CW_{\min}$-1) instead than in the range (0, $CW_{\min}$). This can be accounted in the DCF model by simply setting, in Eq. (31), $W_0$=$CW_{\min}$ instead of $W_0$=$CW_{\min}$+1. Finally, since, according to the standard specification, a contending station will wait for an ACK_Timeout greater than an EIFS, before reattempting to transmit[10], then

$$\overline{T_c} = T_c + \sigma \tag{41}$$

The throughput shall be computed as in the usual case, but with the new values defined in Eqs. (39), (40) and (41):

$$S = \frac{P_{success}\,\overline{E[P]}}{P_{idle}\,\sigma + P_{success}\,\overline{T_s} + (1 - P_{idle} - P_{success})\,\overline{T_c}} \tag{42}$$

We remark that, in practical cases (e.g., the 802.11b parameters – we recall that $CW_{\min}$=31 in this case), the difference between the results computed via Eq. (42) and that computed via the more approximate expression in Eq. (37) are negligible.

#### 4.3.3.2 Delay Performance

In saturation conditions, the total delay experienced by a frame is not meaningful. The time elapsed from the time instant when the frame is inserted in the transmission buffer to the time instant when it is successfully transmitted depends on how long the system has remained in saturation conditions, and consequently how congested the transmission buffer has become (this in turn depends on how greater the offered load is with respect to the saturation throughput bound).

Nevertheless, it is instructive to quantify the average *access* delay D, defined as the time elapsed between the time instant when the frame is put into service - i.e., it becomes

[10] The Ack_Timeout is specified in the Annex C (For formal description of MAC operation, see details of the Trsp timer setting on page 346) as:

Ack_Timeout = CTS_Timeout = aSifs + Duration(Ack) + PLCPHeader + PLCPPreamble + aSlotTime.

According to this value, a station involved in a collision will be able to access the channel only a DIFS after the Ack_Timeout, and thus (in the assumption of 1 Mbps control rate) only a slot-time after the end of an EIFS for monitoring stations. Therefore, we conclude that the extra slot after the end of an EIFS will not be used by any station (either involved in a collision as well as other stations monitoring the channel). Of course, in the assumption that a DIFS is employed after a collision, then the fact that an Ack Timeout is greater than a DIFS is even more evident.

head-of-line (HOL) - and the instant of time the frame terminates a successful delivery. Under the assumption of no retry limits, i.e., that all the HOL frames are ultimately delivered, this computation is straightforward. In fact, we may rely on the well known Little's Result, which states that, for any queueing system, the average number of customers in the system is equal to the average experienced delay multiplied by the average customer departure rate. The application of Little's result to our case yields:

$$D = \frac{N}{S / E[P]} \tag{43}$$

In fact, under the assumption that no frames are lost because of retry limit (i.e., R=infinity), each of the $N$ stations is contending with an HOL frame. Moreover, $S/E[P]$ represents the throughput S measured in frames per seconds, and thus represents the frame departure rate from the system.

The delay computation is more elaborate when a frame is discarded after reaching a predetermined maximum number of retries R. In fact, in such a case, a correct delay computation should take into account only the frames successfully delivered at the destination, while should exclude the contribution of frames dropped because of frame retry limit (indeed, the delay experienced by dropped frames would have no practical significance).

To determine the average delay in the finite retry case, we can still start from Little's Result, but we need to replace $N$ in Eq. (43) with the average number of HOL frames that will be successfully delivered. This value is lower than the number of competing stations, as some of the competing frames will be ultimately dropped. Thus, Eq. (43) can be rewritten as follows:

$$D = \frac{N\left(1 - P\{LOSS\}\right)}{S / E[P]} \tag{44}$$

where $P\{LOSS\}$ represents the probability that a *randomly* chosen HOL frame will be ultimately dropped. Let us now randomly pick an HOL frame among the $N$ contending ones. Such an HOL frame can be found in any of the $i$=0,…,$R$ possible backoff stages. The probability that a random frame is found in backoff stage $i$ has been expressed in Eq. (26), and can be rewritten in terms of known values $p$, $\tau = P\{TX\}$ and $\beta_i$ by means of the equalities in Eqs. (29) and (30):

$$P\{s = i\} = P\{TX\}\frac{P\{s = i \mid TX\}}{P\{TX \mid s = i\}} = \tau\frac{(1-p)p^i}{1-p^{R+1}}(1+\beta_i) \tag{45}$$

By conditioning on the backoff stage $i$, $P\{LOSS\}$ can be now computed as:

$$
\begin{aligned}
P\{LOSS\} &= \sum_{i=0}^{R} P\{LOSS \mid s=i\} \cdot P\{s=i\} = \\
&= \sum_{i=0}^{R} p^{R+1-i} \cdot \tau \frac{(1-p)p^{i}}{1-p^{R+1}} (1+\beta_i) = \\
&= \frac{p^{R+1}}{1-p^{R+1}} \tau (1-p) \sum_{i=0}^{R} (1+\beta_i)
\end{aligned}
\tag{46}
$$

The average access delay expression is now found by substituting Eq. (46) into Eq. (44). In the derivation of Eq. (46), we have made use of the fact that the probability that a frame found in backoff stage $i$ is ultimately dropped, is given by the probability that it first reaches the backoff stage $R$ (i.e., it collides for $R$-$i$ times), and then it also collides during the last transmission attempt. Hence $P\{LOSS|s=i\}=p^{R+1-i}$.

The average delay expression can be further simplified. From Eq. (33), the probability of a successful transmission, expressed in Eq. (10), can be rewritten as:

$$
P_{success} = N\tau(1-p) \tag{47}
$$

Hence, recalling that the throughput $S$ was computed in Eq. (12) as the probability of successful transmission multiplied by the ratio $E[P]/E[slot]$,

$$
\begin{aligned}
D &= \frac{N(1-P\{LOSS\})}{S/E[P]} = \\
&= \frac{N - \frac{p^{R+1}}{1-p^{R+1}} N\tau(1-p) \sum_{i=0}^{R} (1+\beta_i)}{S/E[P]} = \\
&= \frac{N}{S/E[P]} - E[slot] \frac{p^{R+1}}{1-p^{R+1}} \sum_{i=0}^{R} (1+\beta_i)
\end{aligned}
\tag{48}
$$

This final expression has an elegant intuitive interpretation[11]. From Little's Result, the first term represents the average inter-departure time between two *successfully* delivered frames from a same station. This differs from the average access delay as, between two successful transmissions, a number of dropped frames may occur. Now, $p^{R+1}$ represents the probability that a new frame entering the system (i.e., placed in HOL position) will be dropped (note the difference with $P\{LOSS\}$, which instead represents the probability that a randomly chosen frame, among the contending ones, is lost). Assuming independent frame dropping, the *average* number of dropped frames between two successful delivery is thus given by the ratio $p^{R+1}/(1-p^{R+1})$. A dropped frame will be forced to cross all the backoff stages, from stage 0 to stage $R$, and in each stage $i$ it will spend, in average, $(1+\beta_i)$ slots. Hence the average delay for a successfully transmitted frame is given by the average inter-

[11] This neat interpretation was suggested by A.C. Boucouvalas, P. Chatzimisios, and V. Vitsas in a private communication to one of the authors of this chapter. The delay expression given in Eq. (48) was first derived in their recent work with a technical approach different from that presented here [23].

departure time between successful frames, namely, the first term in Eq. (48), minus the time spent by dropped frames. Hence, we might have directly written Eq. (48) from this intuitive reasoning, with no need to provide any formal derivation at all!

### 4.3.4 Non-Ideal Channel Conditions

The performance analyses described in the previous sections were based on the assumption of ideal channel conditions. In this section, we show how, with suitable simplifying assumptions, it can be extended to account for frame corruption.

Let us first remark that the thorough evaluation of the error probability encountered by a frame transmission would require a detailed investigation dealing with physical layer transmission details, fading channel modelling, and interference/capture issues. Such an investigation is out of the scopes of the present section. However, in the simplifying assumption that all frames are subject of the same, known, corruption probability, the analysis becomes straightforward.

Let $\zeta$ be the probability that a transmitted frame is corrupted because of noisy channel conditions, and assume, for convenience of presentation, that errors may occur only on the transmitted MPDU (and not on control frames such as ACK, RTS and CTS – the extension of the analysis to account for control frame errors is immediate). Since the transmitting station will not receive an explicit acknowledgement, it will increment its backoff stage regardless of the fact that a collision occurred on the channel, or the frame was simply corrupted by channel noise. Hence, in case of transmission impairments, the conditional collision probability $p$, defined in the previous section as the probability that a transmitted frame collides, now represent the union of the events i) the frame collided, and ii) the frame was corrupted. In formulae:

$$p = 1 - (1 - \zeta)(1 - \tau)^{N-1} \tag{49}$$

As usual, $\tau$ represents the probability that a station transmits in a randomly chosen slot. With this new definition of $p$, it is immediate to realize that the computation of $\tau$ is not affected. Thus, Eq. (32) still holds and can be jointly solved with Eq. (49) to obtain numerical expressions for $p$ and $\tau$.

Some additional care is required to compute the saturation throughput. In fact, it is necessary to determine the proper probabilities of the various events that may occur on the channel, events which now include the case of frame corruption. We can express the throughput $S$ as:

$$S = \frac{(1-\zeta)P_{success}E[P]}{P_{idle}\sigma + (1-\zeta)P_{success}T_s + \zeta P_{success}T_e + (1 - P_{idle} - P_{success})T_c} \tag{50}$$

where the probability $P_{success}$ is still given by Eq. (10), but this time it represents the probability that a single frame is transmitted in a slot, i.e., it does not contend with other frames. Of course the frame will be successfully delivered only if it is transmitted alone in a slot, and it results not corrupted, i.e., with joint probability (1-$\zeta$) $P_{success}$.

In Eq. (50), a new value $T_e$ is introduced to account for the duration of a period in which no other stations can access the channel because a corrupted transmission is being occurring on the channel. A listening station which detects the transmitted frame as corrupted, will wait for an EIFS time. Thus:

$$T_e = T_{MPDU} + EIFS \tag{51}$$

Note that some of the surrounding stations may correctly detect the frame. Hence, they will be able to read the duration field in the MAC header of the transmitted frame, and set the NAV accordingly. In other words, under the assumption that the ACK is transmitted at 1 Mbps, $T_e$ will result in the same value regardless of the fact that a listening station sees the transmitted frame as a correct or corrupted one.

## 4.4 MAC Enhancements for QoS Support

In this section, we present the 802.11e MAC [3] for QoS provisioning. The IEEE 802.11e defines a single coordination function, called the hybrid coordination function (HCF). The HCF combines functions from the DCF and PCF with some enhanced QoS-specific mechanisms and QoS data frames. Note that the 802.11e MAC is backward compatible with the legacy MAC, and hence it is a superset of the legacy MAC. The HCF is composed of two channel access mechanisms: (1) a contention-based channel access referred to as the *enhanced distributed channel access* (EDCA), and (2) a controlled channel access referred to as the *HCF controlled channel access* (HCCA). The HCF sits on top of the DCF in the sense that the HCF utilizes and honors the CSMA/CA operation of the DCF. In a *QoS-enabled IBSS* (QIBSS), only the EDCA can be used since the HCCA relies on the AP for the channel control. Since we are considering the IBSS operation here, we will limit ourselves to the EDCA operation in the following.

### 4.4.1 IEEE 802.11e EDCA

The EDCA is designed to provide differentiated, distributed channel accesses for frames with 8 different user priorities (UPs) (from 0 to 7) by enhancing the DCF. Each MSDU from the higher layer arrives at the MAC along with a specific user priority value. Each QoS data frame also carries its user priority value in the MAC frame header. An 802.11e station shall implement four channel access functions, where a channel access function is an enhanced variant of the DCF, as shown in Figure 4.9. Each frame arriving at the MAC with a user priority is mapped into an access category (AC) as shown in Table 4.2, where one of the four channel access functions is used for each AC. Note the relative priority of UP 0 is placed between 2 and 3. This relative priority is rooted from IEEE 802.1d bridge specification [4].

Basically, a channel access function uses AIFS[12][AC], $CW_{min}$[AC], and $CW_{max}$[AC] instead of DIFS, $CW_{min}$, and $CW_{max}$, of the DCF, respectively, for the contention to transmit a frame belonging to access category AC. AIFS[AC] is determined by

[12] AIFS: Arbitration Interframe Space, referring to IEEE 802.11e MAC [3].

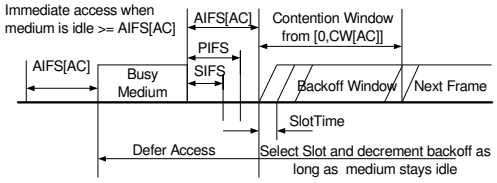


**Figure 4.9: IEEE 802.11e EDCA channel access.**

$$AIFS[AC] = SIFS + AIFSN[AC] \cdot SlotTime, \tag{52}$$

where AIFSN[AC] is an integer greater than one. Figure 4.9 shows the timing diagram of the EDCA channel access. One big difference between the DCF and EDCA in terms of the backoff countdown rule is as follows: the first countdown occurs at the end of the AIFS[AC] interval. Moreover, at the end of each idle slot interval, either a backoff countdown or a frame transmission occurs, but not both. Note that according to the legacy DCF, the first countdown occurs at the end of the first slot after the DIFS interval, and if the counter becomes zero during a backoff process, it transmits a frame at that moment.

Figure 4.10 shows the 802.11e MAC with four channel access functions, where each function behaves as a single enhanced DCF contending entity. Each channel access function has its own AIFS and maintains its own backoff counter. Accordingly, these four channel access functions contend for the medium in parallel independently. The channel access function finishing the backoff the earliest transmits its pending frame into the medium, and the rest suspend their backoff process until the medium becomes idle again. However, when there is more than one channel access function finishing the backoff at the same time, the collision is handled in a virtual manner. That is, the highest priority frame among the to-be colliding frames is chosen and transmitted, and the others perform a backoff with increased CW values.

Apparently, the values of AIFS[AC], $CW_{min}$[AC], and $CW_{max}$[AC], referred to as the EDCA parameters, play the key role for the differentiated channel access among different user priority (or AC more accurately speaking) frames. Basically, the smaller AIFS[AC], $CW_{min}$[AC], and $CW_{max}$[AC], the shorter the channel access delay for access category AC, and hence the more bandwidth share for a given traffic condition. In the infrastructure mode, these EDCA parameters can be determined and announced by the AP via beacon frames. However, there is no AP in an IBSS, and hence the default parameters as shown in Table 4.3 are used in an IBSS [3]. The parameters aCWmin and aCWmax in the table refer to the $CW_{min}$ and $CW_{max}$ values for different PHYs, respectively, e.g., the values found in Table 4.1.

**Table 4.2: User priority to access category mappings.**

| Priority | User Priority (UP) | Access Category (AC) | Designation (Informative) |
|---|---|---|---|
| Lowest | 1 | AC_BK | Background |
| | 2 | AC_BK | Background |
| | 0 | AC_BE | Best Effort |
| | 3 | AC_BE | Best Effort |
| | 4 | AC_VI | Video |
| | 5 | AC_VI | Video |
| | 6 | AC_VO | Voice |
| Highest | 7 | AC_VO | Voice |

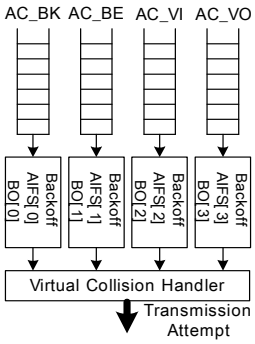


**Figure 4.10: Four channel access functions for EDCA.**

**Table 4.3: Default EDCA Parameter Set.**

| AC | $CW_{min}$ | $CW_{max}$ | AIFSN | TXOP Limit | | |
|---|---|---|---|---|---|---|
| | | | | DS-CCK[13] | Extended Rate/OFDM[14] | Other PHYs |
| AC_BK | aCWmin | aCWmax | 7 | 0 | 0 | 0 |
| AC_BE | aCWmin | aCWmax | 3 | 0 | 0 | 0 |
| AC_VI | (aCWmin+1)/ 2-1 | aCWmin | 2 | 6.016 ms | 3.008 ms | 0 |
| AC_VO | (aCWmin+1)/ 4-1 | (aCWmin+1) /2-1 | 2 | 3.264 ms | 1.504 ms | 0 |

One distinctive feature of the 802.11e is the concept of transmission opportunity (TXOP), which is an interval of time when a particular station has the right to initiate transmissions. During a TXOP, there can be multiple frame exchange sequences, separated by SIFS, initiated by a single station. A TXOP can be obtained by a successful EDCA contention, and it is referred to as an EDCA TXOP. The duration of a TXOP is determined by another EDCA parameter, called TXOP limit. This value is determined for each AC, and hence is represented as TXOPLimit[AC]. The default values of the TXOP limits are also shown in Table 4.3.

During an EDCA TXOP, a station is allowed to transmit multiple MSDUs of the same AC with a SIFS time gap between an ACK and the subsequent frame transmission. Figure 4.11 shows the transmission of two QoS data frames of user priority UP during an EDCA TXOP, where the whole transmission time for two data and ACK frames is less than the EDCA TXOP limit. The multiple consecutive frame transmissions during a TXOP can enhance the communication efficiency by reducing unnecessary backoff procedures.

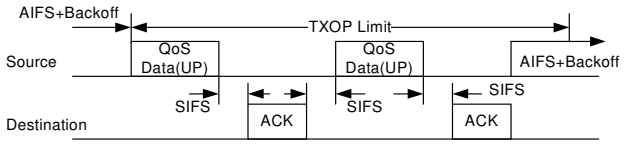


**Figure 4.11: EDCA TXOP operation timing structure.**

[13] Referring to IEEE 802.11b PHY [6].
[14] Referring to IEEE 802.11a [5] and 802.11g PHYs [7].

### 4.4.2 Further QoS Enhancement for Ad-Hoc Networks

As explained above, the current 802.11e EDCA for the ad-hoc mode can provide the differentiated channel access for different user priority frames. However, its capability as defined currently seems to be limited since it relies on the fixed default EDCA parameters. Without a centralized decision-making entity, e.g., an AP, it sounds like a reasonable choice. However, one can invent better ways for enhanced QoS provisioning in ad-hoc networks.

First, we expect that the 802.11e EDCA in the ad-hoc mode can be further enhanced by incorporating a dynamic EDCA parameter adaptation through the negotiation among the participating stations. Along with a distributed admission control, e.g., the one discussed in [8], such a distributed dynamic EDCA parameter adaptation should be a possibility.

Second, we can also develop a distributed version of a centralized QoS-supporting MAC, e.g., IEEE 802.11e HCCA [3]. By a distributed version, we mean a dynamic and distributed election of a centralized controller. Since the centralized controller does not need to be located in a fixed access point, this kind of centralized MAC can be easily applicable in the ad-hoc network as well. Note that Bluetooth, a wireless personal area network (WPAN) technology, defines this type of MAC [9], and HIPERLAN/2, another WLAN technology, supports this kind of dynamic centralized controller election optionally [10]. As a matter of fact, a similar concept was once discussed for the standardization as part of IEEE 802.11e MAC as well. However, it was not included in the 802.11e specification eventually due to many different reasons including its immaturity and involving complexity [11].

## 4.5 Performance Understanding of IEEE 802.11e EDCA

In order to understand the EDCA prioritization mechanisms, it is useful to describe the channel access in terms of low-level channel access operation.

As shown in Section 4.3.2, whenever all the stations operate in saturation conditions, the DCF channel access can be considered as slotted, since packet transmissions start only in discrete time instants. These instants correspond to an integer number of backoff slots which follow the previous channel activity period plus the DIFS time. By looking only at the time instants in which a packet transmissions can be originated, the granted channel resources can be represented in terms of a sequence of idle slots, corresponding to the backoff slots in which no station accesses the channel, and busy slots, corresponding to the time interval required for the packet transmission (which includes the corresponding acknowledge in case of success) plus the DIFS. Given a channel slot, the DCF fairness property implies that each station has the same probability to start a transmission and to experience a success.

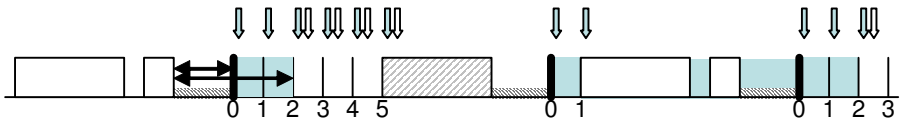


**Figure 4.12: Protected slots in EDCA, in the case of AIFSN[$AC_1$]=AIFSN[$AC_2$]+2.**

The same slotted channel operation can be assumed for describing the channel access occurrence in EDCA. However, the major difference is that the time instants in which the packet transmissions can be originated, which delimit the channel slots, now depend on the minimum AIFS employed by the contending traffic classes. Moreover, because of the different AIFS values, some slots can be accessed only by a subset of the competing traffic classes.

Figure 4.12 shows an example of slotted EDCA channel. The discrete time instants in which channel access can be granted are indicated by arrows, and numbered according to the time elapsed from the last channel activity period. A transmission originated after the minimum AIFS employed in the network[15] belongs to the transmission slot 0, while a transmission originated after $x$ idle backoff slots belongs to the transmission slot $x$. Each arrow represents the probability that a station belonging to a given priority class transmits on the channel. Only two classes are considered in the figure. Since each class employs a different AIFS value, (in the example, the difference among the two values is equal to two backoff slots) some slots can be accessed by one class only (in the example, slots labelled with index 0 and 1). We define these slots, which are shaded and pointed by a single arrow, as *protected*. Note that protected slots occur after each busy slot, and therefore the percentage of protected slots grows as the network congestion increases (this is an immediate consequence of the fact that, with a high number of competing stations, the average number of idle backoff slots, between two consecutive frames transmitted on the channel clearly reduces, and as such the relative amount of protected slots increases).

At the end of each channel access, the stations contend for acquiring the right of the next transmission grant. The contention is based on the comparison of the backoff counter values of each contending station, since the station with the lowest backoff expiration time acquires the right to initiate the next transmission.

*The backoff expiration time does not depend only on the backoff counter value, but also on the specific AIFS setting*, since the resumes of the backoff counters after each channel activity are not synchronous among the stations. In other words, there are two complementary factors which jointly affect the channel access contention. In fact, consider a number of competing stations, and let $b_i$ be the backoff counter value for each station $i$ at the end of a generic channel activity period. If all stations use the same inter-frame space, then the station which would first transmit, and thus "win" the current ongoing contention, would be the one with smaller $b_i$ value. A first way to differentiate performance would then be to configure the backoff operation so that, in average, a group of stations extracts a smaller backoff counter value with respect to the remaining ones (and this can be accomplished by differently setting the $CW_{min}$ value for different classes of stations – see analysis in the next section 4.5.1). However, when different AIFS are employed, if we define with $\delta_i$ the number of extra slots spent while waiting for the station's AIFS to elapse with respect to the minimum possible AIFS value, at the end of a generic channel activity period, the "winning" station would be that one with smaller value $b_i + \delta_i$. The impact of such AIFS differentiation on the channel access contention performance will be tackled in

[15] As described in section 4.5.3.1, thanks to a new specification of the backoff counter decrement, unlike DCF (see section 4.3.3.1), in EDCA the first slot time immediately following a channel activity period is no more implicitly reserved to the station that has just transmitted. This difference will be later on extensively discussed when dealing with EDCA/DCF coexistence.

section 4.5.2. In the following, we refer to this slotted contention resolution model in order to investigate on the effects of the $CW_{min}$ and AIFS differentiation.

### 4.5.1 $CW_{min}$ Differentiation

The generalization of the analysis presented in Section 4.3 to multiple traffic classes is straightforward whenever we only consider the differentiation of the contention windows. In this case, in fact, valid the hypothesis at the basis of the model presented in Section 4.3 it is still valid. According to such hypothesis, the behavior of each station can be summarized by a unique parameter, the channel access probability $\tau$, which is uniform slot by slot. However, the probability $\tau$ is now different depending on the service class the station belongs to, because the different backoff extraction ranges have the effect of increasing/reducing the probability that some stations win the contention against the others. In order to compute the access probability $\tau_k$ for a target station belonging to class $k$, we can still use Eq. (36), by particularizing the specific backoff extraction ranges (given by $W_k = CW_{min}[k]$ and $R_k$):

$$\tau_k = \frac{2(1-2p_k)(1-p_k^{R_k+1})}{(1-2p_k)(1-p_k^{R_k+1}) + W_k(1-p_k)(1-(2p_k)^{R_k+1})} \tag{53}$$

Note that, due to the access probability differentiation, also the collision probability $p_k$ experienced by each station depends on its service class. Specifically, being $C$ the number of different classes, $k \in (1, C)$ the class index, $n_k$ the number of terminals per class, and $\tau_k$ the per-class transmission probability, the per-class conditional collision probability results:

$$p_k = 1 - \frac{\prod_{r=1}^{C}(1-\tau_k)^{n_r}}{(1-\tau_k)} \tag{54}$$

which simply states that the considered station competes with $n_r$ stations of class $r \neq k$ and with $n_k$-1 stations of the same class (excluding the considered one).

The per-class successful access probability is then given by:

$$P_{success}(k) = n_k \tau_k (1-\tau_k)^{n_k-1} \prod_{r=1, r\neq k}^{C}(1-\tau_r)^{n_r} = n_k \tau_k (1-p_k) \tag{55}$$

Thus, the per-class throughput $S_k$ can be computed as:

$$S_k = \frac{P_{success}(k)E[P]}{\prod_{r=1}^{C}(1-\tau_r)^{n_r}\sigma + \sum_{r=1}^{C}P_{success}(k)T_s + \left(1-\prod_{r=1}^{C}(1-\tau_r)^{n_r} - \sum_{r=1}^{C}P_{success}(k)\right)T_c} \tag{56}$$

where the numerator represents the average number of payload bits transmitted by stations belonging to class $k$ in each slot, and the denominator represents the average slot duration,

which is common for all the classes. The ratio between the aggregated throughput perceived by different service classes, in the case of fixed payload size, is simply expressed by the ratio among the successful access probabilities:

$$\frac{S_j}{S_k} = \frac{P_{success}(j)}{P_{success}(k)} \tag{57}$$

Whenever the number of contending stations is low and the per-class collision probability results negligible, the successful access probability can be approximated by $P_{success}(k)=n_k\tau_k$. In the same hypothesis of negligible collision probability $p_k$ the per-class access probability $\tau_k$ is:

$$\tau_k \approx \frac{2}{1+W_k} \tag{58}$$

In this case the throughput ratio among the classes can be immediately related to the minimum contention window settings:

$$\frac{S_j}{S_k} \approx \frac{n_j\tau_j}{n_k\tau_k} = \frac{n_j(1+W_k)}{n_k(1+W_j)} \approx \frac{n_j}{W_j}\cdot\frac{W_k}{n_k} \tag{59}$$

where it is evident that the throughput repartition among the stations results proportional to the inverse of the minimum contention window.

Figure 4.13 compares our approximated evaluation of the throughput repartition (points) with some results obtained via simulation (lines). We used the ns-2 simulator platform with custom-made 802.11e extensions. We assume that an equal number of $n_k=N$ high priority stations share the channel with $n_j=N$ low priority stations. In this case the throughput repartition $S_j/S_k$ is approximated by $W_k/W_j$ which does not depend on the network congestion status. In fact, from the figure we observe that the points match very well both the *N=2* and *N=10* simulation curves, which are very close each other. We can conclude that *the throughput repartition due to the $CW_{min}$ differentiation is almost independent on the network load.* Thus, as the network congestion increases, high priority and low priority stations suffer proportionally of the throughput degradation due to the collision probability grow.

### 4.5.2 AIFS Differentiation

AIFS differentiation is motivated by a completely different (and somewhat more complex) physical rationale. Rather than differentiating the performance by changing the backoff structure (through different settings of the $CW_{min}$ and $CW_{max}$ parameters), the idea is to reserve channel slots for the access of higher priority stations. In fact, as in the case considered in Figure 4.12, when some stations employ different AIFS values, there exists a period of time in which the stations with shorter AIFS value (namely, the higher priority stations) may access the channel, while the stations with longer AIFS (lower priority stations) are prevented from accessing the channel.

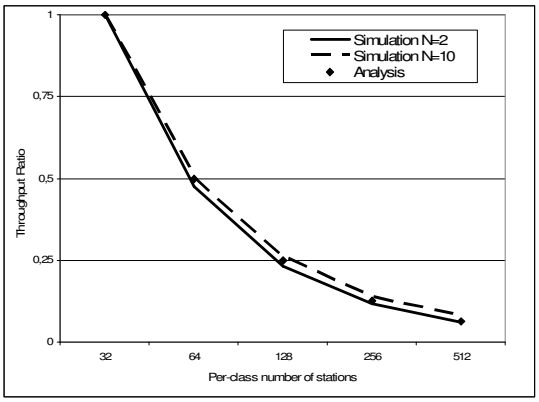


**Figure 4.13: Throughput ratio among two service classes with $CW_{min}$ equal to 32 and $CW_{min}$ equal to the x-axis.**

A fundamental issue of the AIFS differentiation is that protected slots occur after every busy channel period. This implies that the percentage of protected slots significantly increases as long as the network congestion increases. In fact, a greater number of competing stations implies that the average number of slots between consecutive busy channel periods reduces, and thus the fraction of protected slots over the total number of idle slots gets larger.

We already observed that the number of stations which can access each channel slot is not constant slot by slot, but depends on the time elapsed from the previous transmission. This means that in presence of AIFS differentiation the hypothesis about uniform per-slot collision probability is no more valid. Thus, the protocol analysis cannot be a simple generalization of the DCF one, as in the case of $CW_{min}$ differentiation.

Indeed, several approaches have been recently proposed in literature to solve this issue [24, 28, 29, 30], with different levels of complexity and accuracy. Since the analytical approaches employed in order to duly modeling AIFS differentiation are somewhat complex and formally cumbersome, we refer the reader interested in the AIFS modeling details, and in the derivation of absolute throughput values, to such specialized literature works. Instead, in this book chapter, we propose a much simpler and intuitive modeling approach targeted to derive relative throughput figures (i.e. the ratio between the throughput figures expected by different classes). Although limited to operate under the assumption of limited collision probability, nevertheless the proposed model has the advantage of being straightforward and providing an immediate physical understanding of why AIFS differentiation is effective in differentiating performance between distinct traffic classes.

For convenience of presentation, let us restrict to the case of two service classes *j* and *k* which do employ the same contention window parameters, but which differentiate each other in terms of AIFS values. More specifically, let the AIFS setting of stations belonging to class *j* be greater than the AIFS setting of class *k* stations of an integer number $\delta_j$ of backoff slots. Obviously, class *j* stations will experience a lower channel access priority than class *k* ones, due to the longest AIFS setting.

Let us now assume that the stations belonging to both classes experience a marginal collision probability. Owing to this approximation[16], we can conclude that the transmission probabilities $\tau_j$ and $\tau_k$ of stations belonging to these two classes are the same, and that such a transmission probability is given by

$$\tau_k \approx \tau_j \approx \frac{2}{1+W} = \tau \tag{60}$$

Even if the different AIFS settings are assumed not to change the channel access probability $\tau$, nevertheless, stations belonging to different classes will not access the same set of slots (as pointed out by the arrows in Figure 4.12), this yielding to prioritization of class *k*. In fact, at the end of a generic transmission occurring on the channel, class *j* stations will be able to access the channel only in the non-protected slots, i.e. when at least $\delta_j$ empty backoff slots have elapsed. Let now $n_k$ and $n_j$ be, respectively, the number of competing high priority and low priority stations. The probability that no high priority station accesses the channel during the $\delta_j$ protected slots is given by the probability that neither of the $n_k$ high priority class *k* stations transmit in any of the $\delta_j$ protected slots, i.e.,

$$(1-\tau)^{n_k \delta_j} \tag{61}$$

In the non-protected slots, since all stations are assumed to employ the same transmission probability, all the stations will have the same probability to win the contention, regardless of the service class they belong to. Therefore, the probability that the next successful transmission will be generated from a station in class *j* is readily given by

$$\frac{n_j}{n_k + n_j} \tag{62}$$

Neglecting collisions, we can hence trivially approximate the throughput repartition among the two priority classes in terms of ratio among the contention winning probabilities, i.e.:

$$\frac{S_j}{S_k} \approx \frac{\frac{n_j}{n_j + n_k}(1-\tau)^{n_k \cdot \delta_j}}{1 - \frac{n_j}{n_j + n_k}(1-\tau)^{n_k \cdot \delta_j}} \tag{63}$$

Figure 4.14 compares our approximated evaluation of the throughput repartition (points) with some results obtained via simulation (lines). Also in this case, we assume that

[16] Different traffic classes will in fact experience different transmission probability values, if collision is not negligible. In fact, low priority stations (class *j*) will more frequently experience collisions, when compared to high priority stations (class *k*), and hence the exponential backoff operation will increase the average contention window value employed throughout the contention process. This will in turns reduce the transmission probability $\tau_j$ when compared with $\tau_k$.

an equal number of $n_k=N$ high priority and $n_j=N$ low priority stations share the same channel. From the figure we observe that, despite its simplicity, our approximation is quite accurate in capturing the experienced throughput ratio. As expected, the accuracy slightly degrades as $N$ increases because of the emerging occurrence of non negligible collision probabilities, although the difference with simulation results remain fairly limited even when as much as 10 stations ($N$=5) do compete against each other.

The figure clearly highlights that the effectiveness of AIFS differentiation significantly depends on the number of competing stations, i.e., the network congestion status. For example, two protected slots correspond to a throughput ratio of about 65% in the case of $N$=2 and to a throughput ratio of about 37% (i.e., even if the same number of stations per priority class are competing, high priority stations share almost ¾ of the total available channel capacity) in the case of $N$=5. We can conclude that *the throughput repartition due to the AIFS differentiation is strongly affected by the high priority load: the higher the load, the greater the effectiveness of AIFS differentiation in protecting the high priority class.* This results in a complementary behavior if compared with the $CW_{min}$ differentiation effect.

### 4.5.3 Coexistence of EDCA AC_BE and Legacy DCF Stations

In this section we try to clarify the rationale of the AC_BE default settings suggested in the standard through intuitive insights rather than through formal derivations [31]. Since EDCA is backward compatible with standard DCF, we expect that the best effort traffic category is somehow equivalent to the legacy DCF traffic. However, from Table 4.3, we see that the access parameters have some differences. Despite of the same minimum and maximum contention window value, the inter-frame time value for the AC_BE is higher than a DIFS (we recall that a DIFS is equal to an AIFS with AIFSN=2).

Figure 4.15 shows throughput results in a scenario in which $N$ legacy DCF stations share the channel with the same number $N$ of EDCA stations. Curves with the same symbol refer to the same simulation. The bold lines represent the aggregate EDCA throughput, while the thin lines represent the aggregate DCF throughput. EDCA stations have been configured with the standard DCF backoff parameters ($CW_{min}$=31 and $CW_{max}$=1023). The packet size has been fixed to 1500 bytes (Ethernet MTU) and the retransmission limit is set to 7 for all the stations. Control frames are transmitted at a basic rate equal to 1 Mbps, while the MPDU is transmitted at 11 Mbps. Unless otherwise specified, these settings have been maintained in all the simulations. We measured performance in saturation conditions. Although this assumption is not realistic for real-time application, it represents a very good representation of elastic data traffic and an interesting case study to derive the limit performance, i.e., the maximum amount of bandwidth that AC_BE can obtain sharing the channel with best effort DCF stations. From the figure we see that in the case AIFSN=2 the EDCA stations receive much more resources than DCF stations, while in the case AIFSN=3 they achieve performance close to that of legacy DCF stations.

This counter-intuitive result confirms that the default settings have been chosen in order to guarantee backward compatibility with the DCF. However, we need a detailed analysis of the channel access operations in DCF and EDCA to fully understand how this compatibility is provided.

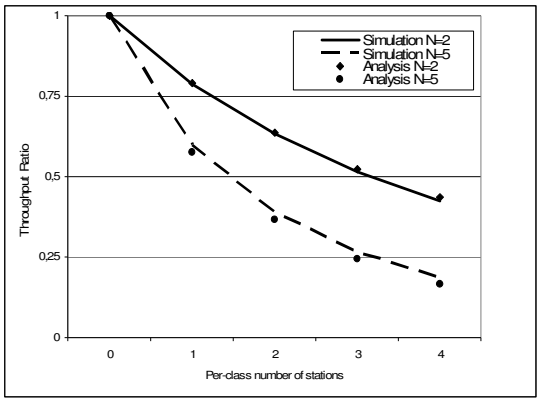


**Figure 4.14: Throughput ratio of two service classes with $\delta_k$ equal to 0 and $\delta_j$ equal to the x-axis.**

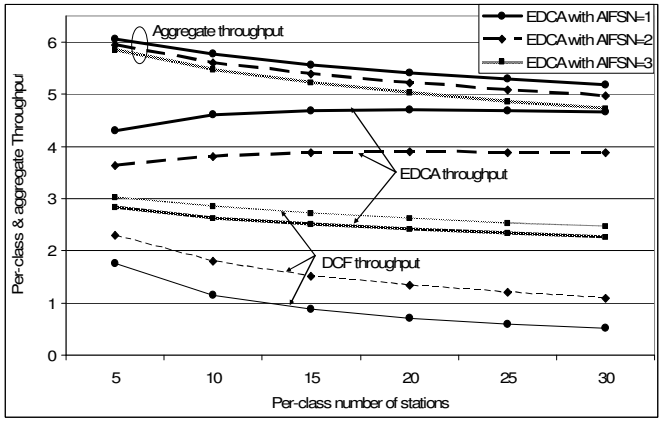


**Figure 4.15: DCF versus EDCA throughput with AIFS Differentiation.**

### 4.5.3.1 Backoff Counter Decrement Rules

EDCA slightly differs from DCF in terms of how the backoff counter is managed (decremented, frozen, resumed). However, such apparently minor difference (which might perhaps appear as a technicality) indeed has some important consequences in terms of performance of EDCA access categories, especially when they compete with legacy DCF stations.

In standard DCF, the backoff counter is decremented at each idle slot-time, frozen during channel activity periods, and resumed after the medium is sensed idle again for a DIFS interval. This implies that a legacy DCF station, after a DIFS, resumes the backoff counter to the discrete value the station had at the instant of time the busy channel period started. An illustrative example is shown in Figure 4.16. Here, a busy channel period (i.e. a transmission from one or more other stations) starts while the backoff counter of the considered DCF station is equal to 4. This value will be frozen during the busy channel period, and will be resumed, again to the value 4, only a DIFS after the end of the busy

period. As a consequence, it will be decremented to the value 3 only a slot after the DIFS. In EDCA, the backoff counter is also decremented at every idle slot-time and frozen during channel activity periods, but it is resumed one slot-time before the AIFS expiration. This means that, when the AIFS timer elapses, the backoff counter will result to be already decremented of one unit.

Moreover, since a single MAC operation per-slot is permitted (backoff decrement or packet transmission, see [2], Clause 9.9.1.3), when the counter decrements to 0, the station cannot transmit immediately, but it has to wait for a further backoff slot if the medium is idle, or for a further AIFS expiration if the medium is busy. In order to understand how these different rules affect the channel access probability, refer to the example shown in Figure 4.16.

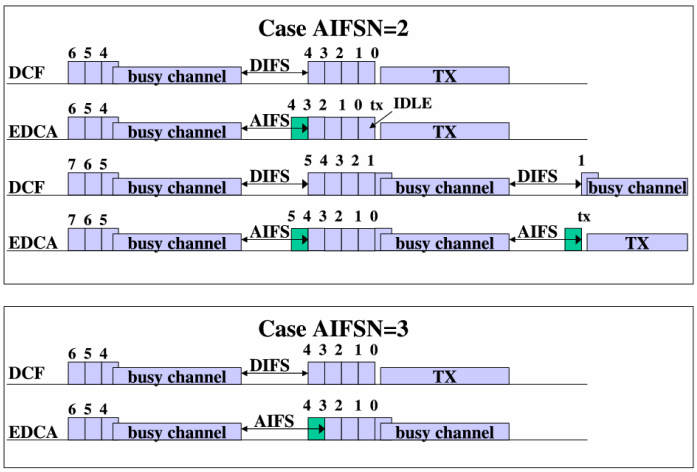


**Figure 4.16: Backoff counter management in EDCA and DCF.**

Let us first focus on the case AIFSN=2 (top figure), which corresponds to using an AIFS equal to the DCF DIFS. In the example, two stations encounter a busy channel period with the same backoff counter value. However, at the end of the channel activity, we see that the DCF station resumes its counter to a value equal to the frozen value (4 in the example), while the EDCA station resumes and decrements its counter. In the case of a single busy channel period encountered during the backoff decrement process, this difference will be compensated by the fact that the EDCA station will have to wait for an extra slot, i.e., unlike the DCF station, it will transmit in the slot following the one in which the backoff counter is decremented to 0 (as illustrated in Figure 4.16 for the top EDCA station).

However, in the presence of several busy channel periods encountered during the backoff decrement process (which is very likely to happen in the presence of several competing stations), the EDCA station will gain a backoff counter decrement advantage for every encountered busy period with respect to the DCF station. This implies that, for an AIFSN equal to a DIFS, the EDCA station has an advantage over DCF, as we observed in Figure 4.15, and this advantage grows as the number of contending stations increases.

Indeed, Figure 4.16 shows that there is also a second reason why, with same inter-frame space AIFSN=2, the EDCA station gains priority over DCF stations. In fact, as

shown in the figure, an EDCA station may actually transmit in the slot immediately following a busy channel period (it is sufficient that the busy channel period was encountered while the backoff counter was equal to 0 – last case in Figure 4.16 with AIFSN=2). Conversely, a DCF station cannot de-freeze a backoff counter value equal to zero. Thus, the only case in which it can access the slot immediately following a busy period is when it extracts a new backoff counter, after a successful transmission, exactly equal to 0.

In order to synchronize the EDCA and DCF backoff decrements, it appears appropriate to set AIFSN=3. In this case, as we can see in the bottom part of Figure 4.16, although the EDCA station has a higher inter frame space, after each busy slot the backoff evolution of the two target stations is the same. However, since the EDCA station has to wait for a further channel slot after the counter expiration, the access probabilities of the two stations does not coincide, since, for a given extraction, the EDCA station has always to wait for a slot more than the DCF station. However, this results in just a slightly higher access probability for the DCF station, which justifies the slightly higher throughput performance observed in Figure 4.15 in the comparison between DCF stations and EDCA stations with AIFSN=3.

#### 4.5.3.2 Analysis of AC_BE Default Settings

The throughput results shown in Figure 4.15 show that, for the same contention window parameters, EDCA throughput performance are similar to that of legacy stations with AIFSN=3 (i.e., the EDCA AC_BE Access Category, see Table 4.3) rather than to a legacy DIFS (i.e., AIFSN=2). The discussion carried out in the previous section has provided a qualitative justification.

Goal of Figure 4.17 is to back-up the previous qualitative explanation with quantitative results. To this purpose, we have numbered slots according to our previous description of the channel access operations. The slot immediately following a DIFS is indexed as slot 0. In the assumption of ideal channel conditions, a successful transmission occurs if, in a transmission slot, only one station transmits; otherwise a collision occurs.

Figure 4.17 reports the probability distribution that a transmission occurs at a given slot, for two different load scenarios: $N$=5, i.e., 5 EDCA stations competing with 5 DCF stations, and $N$=30. Only the first 10 slots are plotted, since most transmissions are originated after very few idle backoff slots. In addition, the figure further details in different colors the probability that a transmission occurring at a given slot results in a collision, in a success for an EDCA station, or in a success for a DCF station.

Figure 4.17 shows that DCF stations are the only ones that can transmit in the slot immediately following the last busy period.

Also, it confirms that a transmission in slot 0 is always successful (as it is originated by a station that has just terminated a successful transmission). Indeed, a transmission in the slot immediately following a busy period is a rare event, since it requires that the station that has just experienced a successful transmission extracts a new backoff counter exactly equal to 0. Thus, the slot 0 is a *protected slot* for the DCF stations, but it is rarely[17]

[17] Quantification is easy: after a successful transmission, a DCF station transmits in the slot 0 only if it exactly extracts a backoff counter equal to 0. This occurs with probability $1/(1+CW_{min}) \approx 3.1\%$. This conditional

granted. The figure also shows that, in the slots with index greater than 0, DCF and EDCA stations experience almost the same success probability, with a negligible advantage for DCF. For example, in the case *N*=5 a DCF success occurs, almost constantly through the various slot indexes, in about the 42.5% of the cases versus the 41% of EDCA, while for *N*=30 these numbers reduce to, respectively, about 32.5% and 31.3% due to the increased probability of collision. The fundamental conclusion is that *by using AIFSN=3, an EDCA station can be set to approximately operate as a legacy DCF station*. With reference to the proposed EDCA parameter settings reported in Table 4.3, we thus conclude that an EDCA station belonging to the Access Category AC_BE will experience similar performance than a legacy DCF station. The above quantitative analysis also justifies why DCF shows a slightly superior throughput performance over EDCA AC_BE, as found in Figure 4.15 under the case of AIFSN=3.

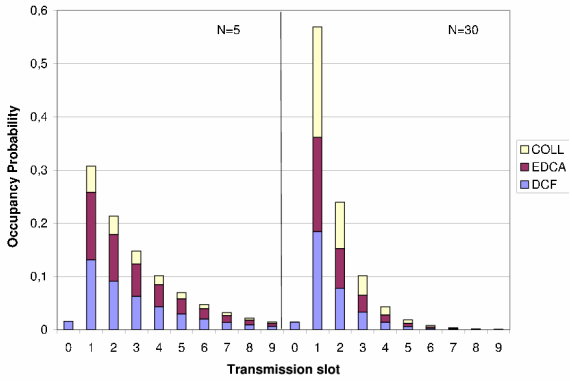


**Figure 4.17: Per-slot occupancy probability - AC_BE versus DCF.**

### 4.5.3.3 AIFSN=2 and Legacy DCF Stations

As shown in Table 4.3, AIFSN=2 is the minimal setting allowed for an EDCA station. The rationale is that both AIFSN=0 and AIFSN=1 are already reserved in the 802.11 standard for, respectively, the SIFS, and for the PIFS. However, as discussed above, the different mechanism employed in EDCA for decrementing the backoff counter suggests that, by using AIFSN=2 (i.e., AIFS=DIFS), an EDCA station is nevertheless expected to gain priority over a legacy DCF station.

This was indeed shown in Figure 4.15, and is strikingly confirmed by Figure 4.18, which, similarly to Figure 4.17, reports the probability distribution that a transmission occurs at a given slot for the scenario of *N* DCF stations competing with *N* EDCA stations configured with AIFSN=2 and standard contention window parameters ($CW_{min}$=31 and $CW_{max}$=1023).

---

probability is consistent with the absolute probability value reported in Figure 4.17 (about a half of this), since about half of the busy periods consist in a successful DCF transmission.

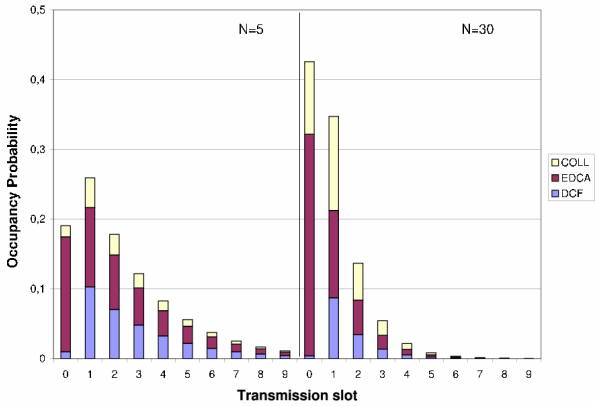


**Figure 4.18: Per-slot occupancy probability - AIFSN=2 versus DCF.**

Figure 4.18 shows, for two different load conditions (*N*=5 and *N*=30), how the channel slots are occupied by the contending stations. From the figure, we see that the slot 0, which, as shown before, is only rarely used by DCF stations, results almost protected for EDCA stations. Channel slots with index higher than 0 are instead accessed by both classes with comparable probability. Figure 4.18 allows us to draw a number of interesting observations. First, the probability of collision in the protected slots (specifically, slot 0) is lower than in the other slots (e.g., for the case *N*=5, a collision in slot 0 occurs only in about 8.5% of the cases, versus an average of 17% in the remaining slots, and these numbers for the case *N*=30 become 24.5% versus 38.5%), due to the reduced number of competing stations. Second, and most interesting, as the network load increases, the probability of accessing low-indexed slots significantly increases. The reason is that the number of slots between two consecutive busy channel periods significantly reduces in high load. But this implies that a large amount of accesses occurs in slots 0 (more than 40% in the case of *N*=30, see Figure 4.18), and thus are almost exclusively dedicated to EDCA stations, with a definite gain in terms of service differentiation effectiveness (as earlier shown in Figure 4.15). As a conclusion, *the usage of AIFSN=2 in EDCA (i.e. AIFS=DIFS) provides a significant priority of EDCA stations over legacy DCF stations*. This is an extremely important fact, as it permits the effective deployment of AIFS differentiation even when DCF stations share the same channel, and thus, apparently, there seems to be no room for AIFS levels intermediate between the inter frame spaces reserved by the standard (SIFS and PIFS), and the legacy DIFS[18].

## 4.6 Conclusions

In this chapter, we have presented a performance study, based on some common modeling approaches, on the IEEE 802.11 DCF and the recent IEEE 802.11e EDCA extensions.

[18] We in fact recall that a legacy DIFS is defined as a SIFS + 2·slot, while a PIFS is defined as a SIFS + 1·slot. Since SIFS and PIFS are parameters reserved for other purposes, the minimum deployable AIFS value coincides with that of a legacy DIFS, and this argument was indeed considered as a limiting factor for EDCA, when competing with legacy 802.11 stations. However, the above discussion show that this is not the case, and that the legacy DIFS is somewhat equivalent – in practice - to an AIFS setting equal to a SIFS + 3·slot.

Regarding the standard DCF, we have first quantified the header and protocol overheads, in case of a single transmitting station. Then, we introduce the concept of saturation throughput, when two or more stations contend greedily for channel access. We have shown that in such conditions, the protocol behavior can be summarized into a single channel access probability $\tau$, and we have derived the system throughput and delay as a function of the number of contending stations and $\tau$. We have illustrated the DCF performance bounds, for the cases of 2-way and 4-way handshake, by showing how the system throughput can be maximized for a given $\tau$ value, which depends on the number of contending stations and on the frame payload. Finally, we concluded the DCF analysis by deriving $\tau$ as a function of the contention window ranges, with elementary conditional probability arguments.

Regarding the EDCA extensions, we have discussed some simple model generalizations able to differentiate the per-station contention window ranges. Then, we focused on the problems arising from modeling the AIFS time differentiation. We proposed an intuitive approximation, able to overcome such modeling complications and to allow an easy tuning of the contention parameters, targeted to guarantee a desired throughput repartition among the service classes. Finally, we presented the system performance for the case of coexistence between EDCA and legacy DCF stations, and justified the equivalence between DCF stations and EDCA best effort class.

## 4.7 References

[1] IEEE 802.11-1999 (R2003), Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, ANSI/IEEE Std 802.11, 1999 Edition (Reaffirmed 2003), June 2003.

[2] IEEE 802-2001, IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture, March 2002.

[3] IEEE 802.11e-2005, Draft Supplement to Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS), Sept. 2005.

[4] IEEE, Part 3: Media Access Control (MAC) bridges, ANSI/IEEE Std. 802.1D, IEEE 802.1d-1998, 1998 edition, 1998.

[5] IEEE 802.11a-1999, Supplement to Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer in the 5 GHZ Band, 1999.

[6] IEEE 802.11b-1999, Supplement to Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-speed Physical Layer Extension in the 2.4 GHz Band, 1999.

[7] IEEE 802.11g-2003, Supplement to Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band, 2003.

[8] Yang Xiao, Haizhon Li, and Sunghyun Choi, "Protection and Guarantee for Voice and Video Traffic in IEEE 802.11e Wireless LANs," in *Proc. IEEE INFOCOM'04*, Hong Kong, March 2004.

[9] Brent A. Miller and Chatschik Bisdikian, *Bluetooth Revealed: The Insider's Guide to an Open Specification for Global Wireless Communications*, Prentice Hall, September 2000.
[10] ETSI TS 101 761-4, Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Data Link Control (DLC) Layer; Part 4: Extension for Home Environment, V1.1.1, June 2000.
[11] Adrian P. Stephens, "AP Mobility Mechanism," IEEE 802.11-02/066r9, March 2002.
[12] Javier del Prado and Sunghyun Choi, "Link Adaptation Strategy for IEEE 802.11 WLAN via Received Signal Strength Measurement," in *Proc. IEEE ICC'03*, Anchorage, Alaska, USA, May 2003.
[13] Daji Qiao, Sunghyun Choi, and Kang G. Shin, "Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs," *IEEE Trans. on Mobile Computing (TMC)*, vol. 1, no. 4, pp. 278-292, October-December 2002.
[14] Jean-Lien C. Wu, Hunh-Huan Liu, and Yi-Jen Lung, "An Adaptive Multirate IEEE 802.11 Wireless LAN," in *Proc. 15th International Conference on Information Networking*, 2001, pp. 411-418.
[15] Ad Kamerman and Leo Monteban, "WaveLAN-II: A High-Performance Wireless LAN for the Unlicensed Band," *Bell Labs Technical Journal*, Summer 1997, pp. 118–133.
[16] Giuseppe Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, March 2000.
[17] H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma, "Performance of Reliable Transport Protocol over IEEE 802.11 Wireless LANs: Analysis and Enhancement", in *Proc. IEEE INFOCOM'02*, 2002.
[18] V. M. Vishnevsky and A. I. Lyakhov, "802.11 LANs: Saturation Throughput in the Presence of Noise", in *Proc. IFIP Networking'02*, Pisa, Italy, 2002.
[19] P. Chatimisios, V. Vitsas, and A. C. Boucouvalas, "Throughput and Delay Analysis of IEEE 802.11 Protocol," in *Proc. the 5th IEEE International Workshop on Network Appliances (IWNA)*, Liverpool, UK, Oct. 2002.
[20] Y. Xiao, "A Simple and Effective Priority Scheme for IEEE 802.11," *IEEE Communications Letters*, vol. 7, no. 2, Feb. 2003, pp. 70-72.
[21] Z. Hadzi-Velkov and B. Spasenovski, "Saturation Throughput – Delay Analysis of IEEE 802.11 DCF in Fading Channels," in *Proc. IEEE ICC'03,* Anchorage, Alaska, May 2003, vol. 1, pp. 121-126.
[22] T.-C. Hou, L.-F. Tsao, and H.-C. Liu, "Throughput Analysis of the IEEE 802.11 DCF Scheme in Multi-Hop Ad Hoc Networks," in *Proc. ICWN'03*, Jun. 2003.
[23] P. Chatzimisios, A. C. Boucouvalas, and V. Vitsas, "IEEE 802.11 Packet Delay – A Finite Retry Limit Analysis," in *Proc. IEEE Globecom'03*, Dec. 2003.
[24] G. Bianchi and I. Tinnirello, "Analysis of Priority Mechanisms Based on Differentiated Inter Frame Spacing in CSMA-CA," in *Proc. IEEE VTC'03-Fall*, October 2003.
[25] Q. Ni, I. Aad, C. Barakat, and T. Turletti, "Modeling and Analysis of Slow CW Decrease for IEEE 802.11 WLAN," in *Proc. IEEE PIMRC'03*, Sept. 2003.
[26] G. Bianchi, L. Fratta, and M. Oliveri, "Performance Evaluation and Enhancement of the CSMA/CA MAC Protocol for 802.11 Wireless LANs," in *Proc. IEEE PIMRC'96*, Taipei, Taiwan, October 1996, pp. 392-396.

[27] F. Calì, M. Conti, and E. Gregori, "Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit," *IEEE/ACM Trans. Networking*, vol. 8, no. 6, Dec. 2000, pp. 785-790.

[28] J. Zhao, Z. Guo, Q. Zhang, W. Zhu, "Performance Study of MAC for service differentiation in IEEE 802.11", in *Proc. IEEE Globecom'02*, Taipei, 2002, pp. 778-782.

[29] J. W. Robinson and T. S. Randhawa, "Saturation Throughput Analysis of IEEE 802.11e Enhanced Distributed Coordination Function," *IEEE Journal on Selected Areas in Communications*, Vol.2 N.5, June 2004, pp. 917-928.

[30] Y. Xiao, "Performance Analysis of IEEE 802.11e EDCF under Saturation Conditions," in *Proc. IEEE ICC'04*, Paris, 2004, pp. 170-174.

[31] G. Bianchi, I. Tinnirello, and L. Scalia, "Understanding 802.11e contention-based prioritization mechanisms and their coexistence with legacy 802.11 stations," *IEEE Network*, Vol. 19, Issue 4, July-Aug. 2005, pp. 28-34.