Saturation throughput analysis of error-prone 802.11 wireless networks

Qiang Ni[1,*, †], Tianji Li[1], Thierry Turletti[2], and Yang Xiao[3]

[1]Hamilton Institute, National University of Ireland Maynooth, Co. Kildare, Ireland
[2]Planete Project, INRIA, Sophia Antipolis, 06902, France
[3]Computer Science Department, University of Memphis, Memphis, TN 38152, U.S.A.

Summary

It is well known that the medium access control (MAC) layer is the main bottleneck for the IEEE 802.11 wireless networks. Much work has been done on the performance analysis of the 802.11 MAC. However, most of them assume that the wireless channel is error-free. In this paper, we investigate the saturation throughput performance achieved at the MAC layer, in both congested and error-prone channels. We provide a simple and accurate analytical model to calculate the MAC throughput with saturated sources. The model is validated through extensive simulation results. Our results show that channel errors have a significant impact on the system performance.

KEYWORDS : IEEE 802.11, wireless LAN (WLAN), medium access control (MAC), distributed coordination function (DCF), saturation throughput, transmission error, collision

1 Introduction

The IEEE 802.11 wireless LAN (WLAN) [1] is a predominant technology for wireless access in local areas: the 802.11b WiFi networks with physical (PHY) layer data rates up to 11Mbps in the 2.4 Ghz frequency band have been widely deployed in hotspots and offices. Furthermore, 802.11a in the 5Ghz band, and 802.11g in dual bands (2.4 Ghz and 5 Ghz), are being deployed to provide higher PHY data rates up to 54 Mbps. To further increase data rate and throughput, the 802.11 working group created a new task group, namely 802.11n, which focuses on the standardization issues of next-generation very high-speed WLANs. This emerging standard is likely to enhance the 802.11a/b/g PHY layer by introducing some advanced channel coding and smart antenna techniques.

In the 802.11 protocol stacks, the medium access control (MAC) layer plays a key role in determining the channel efficiency and quality-of-service (QoS) for upper-layer applications, which receives much attention. The fundamental function to access the wireless medium provided by the MAC layer is called distributed coordination function (DCF). An enhanced DCF mechanism called EDCA, and two polling-based mechanisms (the point coordination function - PCF, and the hybrid coordination function - HCF), were also proposed by the 802.11/802.11e groups. All these mechanisms are based on DCF and are required to be compatible with it. Thus, a thorough understanding of the DCF performance in varying channel conditions is a fundamental research issue for enhancing QoS support and efficiency at the MAC layer.

In the literature, a lot of research efforts have been car-

ried out to model the behavior of DCF with saturated loads in an error-free channel condition (e.g., [2]–[4]). Among them, Bianchi's two-dimensional Markov chain model [2] is a fundamental one. In [3], the seizing effect is added into the model considering that the station that just finished the successful transmission has a better chance to access and seize the channel than others. Another extension is provided in [4] by considering a frame retransmission limit. Furthermore, an analytical model with unsaturated traffic sources can be found in [5].

Recently, several researchers started to analyze the saturation throughput of DCF in error-prone channels [6]–[11]. A Gaussian wireless channel is assumed in which a constant channel bit error rate (BER) is supposed to be known in advance. The channel BER is then added into the Bianchi's model[1]. Note that most of the existing work only considers the error probability of data frames. To the best of our knowledge, [10] is the only one which addresses the error probability of ACK frames. However, the way that they have computed the average time that the channel is sensed busy is not in accordance with the 802.11 standard, e.g., [7]–[8] did not model the impact of EIFS interval in the case of a transmission failure. [11] is different from the above studies, it extends another saturation model in [12] and analyzes the capacity and variability of the MAC protocol in error channel conditions. In summary, a thorough and accurate performance analysis for DCF under both congested and error-prone channel conditions is still missing in the literature.

To this aim, we present a saturation throughput model for the 802.11 DCF scheme in this paper. The 802.11a PHY layer has been chosen as an example to calculate the channel BER although our analytical model can be applied in all kinds of the existing 802.11 PHY layers

---

[1]No techniques were provided in those studies on how to obtain BER.

(e.g., 802.11a, 802.11b, 802.11g and future higher data rate 802.11n). The main contributions of this paper lie in:

- A better understanding and a comprehensive explanation on how the MAC layer handles collisions and transmission errors.

- An analytical model of the 802.11 DCF saturation throughput under congested and error-prone channel conditions.

- A performance investigation of the 802.11 MAC through the analytical model.

The remainder of this paper is organized as follows. Section 2 reviews the 802.11 DCF protocols in congested and error-prone channels. The 802.11a PHY layer is described in Section 3. Section 4 presents our analytical model. Section 5 validates the model by comparing the analytical results with those obtained with simulations. Finally, Section 6 concludes the paper.

2 Collision and error control in DCF

The basic access mechanism of DCF is a *carrier sense multiple access with collision avoidance* (CSMA/CA) scheme together with a binary exponential backoff scheme.

Two kinds of carrier sensing mechanisms are used in DCF: the PHY layer carrier sensing and the MAC layer virtual carrier sensing. The PHY layer carrier sensing is provided by a clear channel assessment function [1], which discovers whether the wireless medium is busy or idle. Then, the PHY layer sends an indication of carrier status to the MAC layer. On the other hand, the MAC layer virtual carrier sensing is optionally used by a transmitting station (STA) to inform all other STAs how long

the channel should be reserved for its transmissions. Before transmitting a frame, a sender fills the duration field in the MAC header to the duration for which it expects to use the medium. Other STAs hearing the transmission can update their local timers called network allocation vectors (NAVs) to this duration. The NAV timer counts down to zero at a uniform rate. When the NAV timer reaches zero, it indicates that the medium is virtually idle.

With DCF, if the medium has been sensed idle for an interval of time equal to a distributed interframe space (DIFS), each STA initializes a counter called *backoff_time* to a random number selected uniformly from the interval [0, CW-1]. CW is called the contention window size. It is doubled after each unsuccessful frame transmission until reaching the maximum value CW_{max}, and it is reset to the minimum value CW_{min} after a successful transmission or if the frame is dropped. The *backoff_time* is decremented by one slot when the medium is sensed idle for the slot, and it is frozen if the medium becomes busy. It resumes after the medium is sensed idle again for a period of DIFS. Once the *backoff_time* reaches zero, the STA is authorized to access the medium. Other STAs hearing the transmission defer their transmissions by adjusting their NAVs. Collisions may occur if multiple STAs start transmissions simultaneously, whereas transmission errors are due to poor channel conditions. In order to notify the sender that the frame has been received successfully, a positive acknowledgement (ACK) is sent out after receiving the data frame. The transmitted data frame and its ACK is separated by a period called the shortest interframe space (SIFS). If an ACK is not received within the period of *ACKTimeout*, most likely because a data frame is corrupted (see Figure 1) or because an ACK frame is corrupted (see Figure 2), the sender assumes that there was a collision. Then it schedules a retransmission by entering the backoff process again until

the maximum *retry limit* is reached.
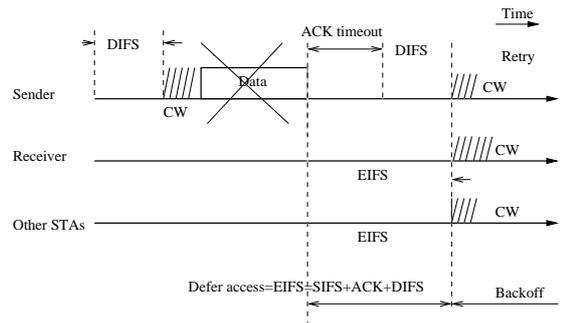
Figure 1: An example of unsuccessful transmission in DCF due to corrupted data
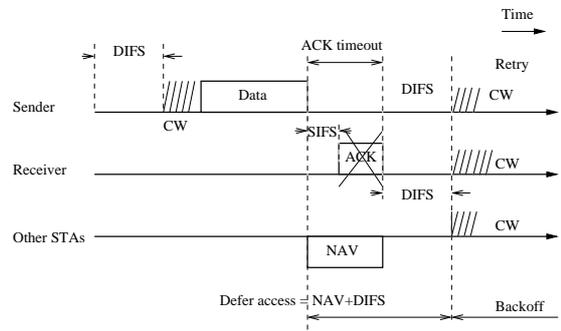
Figure 2: An example of unsuccessful transmission in DCF due to corrupted ACK

It should be pointed out that collisions and transmission errors are not differentiated by the MAC protocol. If a transmitter does not receive an ACK frame, it increases its CW size and retransmits the whole data frame given the *retry limit* is not reached. Indeed, the CW size should be increased only when frame losses are due to collisions in order to decrease congestion, since increasing the CW size is undesirable when losses are due to transmission errors. Random transmission errors can be due to channel fading, path loss, thermal noise, or interferences from other radio sources like Bluetooth devices or microwave ovens. Instead, increasing the CW size in the case of transmission errors may degrade the data throughput and may increase the transmission de-

lays. However, the 802.11 standard cannot distinguish collisions from transmission errors at the MAC layer. If an error is detected in the received data frame by an incorrect frame check sequence (FCS) value, other STAs except the transmitter in the same service area should then use an extended IFS interval (EIFS) which goes beyond the time duration of a DIFS interval as the waiting period. The FCS field in each MAC frame uses a 32-bit cyclic redundancy code (CRC) for the received frame's error-detection. As shown in Figure 1, the EIFS[2] is the sum of DIFS plus SIFS and the time duration for an ACK transmission at the basic data rate [1]. On the other hand, the transmitter will wait for an *ACKTimeout* duration which is usually set equal to ($EIFS - DIFS$). In this way, the transmitter is supposed to have enough time to find out there was a reception error at the receiver side. Then, other STAs in the same service area can defer their transmissions, and reception of a correct frame during the EIFS interval will resynchronize the STA to the actual busy/idle state of the medium. This will terminate the EIFS, and the normal access (using DIFS and backoff) resumes following the reception of that frame [1].

As shown in Figure 2, if a data frame was successfully received but the returned ACK was corrupted, other STAs can still set their NAVs successfully and defer their transmissions to an interval of $NAV + DIFS$. Without receiving an ACK frame in the ACKTimeout period, the transmitter then contends to retransmit the same data frame again after another DIFS and backoff.

To deal with the *hidden terminal* problem, an optional four way hand-shaking technique, known as the Request-ToSend/ClearToSend (RTS/CTS) mechanism, is introduced. Before a data frame transmission, the transmitter sends a short control frame, 20 bytes RTS, and the

receiver replies with a short CTS frame (14 bytes) if it is ready to receive. Once transmitter receives the CTS frame, it transmits the data frame. All other STAs hearing a RTS, a CTS, or a data frame update their NAVs, and will defer their transmissions until the updated NAV timers reach zero.

To regulate the number of retransmissions, two retry counters associated with each MAC frame are set before each transmission: a short retry counter and a long retry counter. For a short frame with length less than or equal to the *RTSThreshold*, the short retry counter is incremented after each retry attempt until this number reaches the *ShortRetryLimit*. The short retry counter should be reset to 0 when the short MAC frame transmission succeeds or the frame is dropped. For the frames longer than the RTSThreshold, retransmissions are done until the number of the attempts reaches the *LongRetryLimit* value or the frame has been successfully transmitted. After reaching either the ShortRetryLimit or the LongRetryLimit values, the frame is discarded.

3   IEEE 802.11a PHY layer

The 802.11a PHY layer uses the orthogonal frequency division multiplexing (OFDM) technology as the modulation scheme [13]. The basic idea of the OFDM modulation is to divide a high-speed binary signal to be transmitted over a number of low data rate sub-carriers. In 802.11a, 52 sub-carriers are introduced, of which 48 sub-carriers carry actual data and 4 sub-carriers are pilots that facilitate phase tracking for coherent demodulation. Each low data rate bit stream is used to modulate a separate subcarrier from one of the channels in the 5 GHz band. Every modulation operation involves translating the data stream into a sequence of symbols. Each symbol may encode a certain number of bits, the number depend-

---

[2]The NAV protection may not be available for other STAs when the transmitted data frame is corrupted.

ing on the modulation scheme. The symbol sequence is then transmitted at a certain rate, called the symbol rate. For a given symbol rate, the data rate is determined by the number of encoded bits per symbol (NBpS). In mobile wireless networks, path loss, fading, and interference cause the variations in the received signal-to-noise ratio (SNR), which also cause the variations in the bit error rate (BER). The lower the SNR, the more difficult it is for the modulation scheme to decode the received signal. Motivated by the observation that for a given SNR, a decrease in PHY data rate by changing modulation modes helps to reduce the BER value, link adaptation schemes[3] are introduced in the 802.11 networks. To support link adaptation, 802.11a defines eight modes with different modulation schemes such as binary phase shift keying modulation (BPSK), quadrature phase shift keying modulation (QPSK), 16-ary quadrature amplitude modulation (QAM), and 64-ary QAM.

Two sub-layers are specified in the PHY layer: the PHY layer convergence procedure (PLCP) sub-layer which performs frame exchanges between the MAC and PHY layers, and the PHY layer medium dependent (PMD) sub-layer which provides actual transmissions and receptions over the wireless medium. Before each transmission, a MAC layer frame is encapsulated into a PLCP sub-layer service data unit (PSDU) frame and then mapped into a PMD sub-layer protocol data unit (PPDU). The PPDU is the actual transmitted unit carried by the 802.11a OFDM technique. The frame formats of PSDU and PPDU in 802.11a are illustrated in Figure 3. Each PHY layer frame includes a PLCP preamble, a PLCP header, a MAC payload, tail bits, and optional padding bits[4]. The PLCP preamble field is used

for synchronization. It consists of 10 short training symbol sequences (0.8 μs each) and two long training symbol sequences (4 μs each). Once the PLCP preamble transmission is started, the PHY layer immediately initiates data scrambling and encoding. The scrambled and encoded data frame should be exchanged between the MAC and PHY layers. The PLCP header, except the SERVICE field, constitutes another OFDM symbol denoted by SIGNAL field in a PPDU frame, which is transmitted in duration of 4 μs with BPSK modulation and rate-1/2 convolutional coding. The 6-bit tail field is used to return the PHY convolutional codec to the zero state. The transmission interval for each OFDM symbol is 4 μs. The 16-bit SERVICE field of the PLCP header and the encapsulated MAC frame (along with six tail bits and pad bits), represented by the DATA field in a PPDU, are transmitted with the data rate specified in the RATE field.
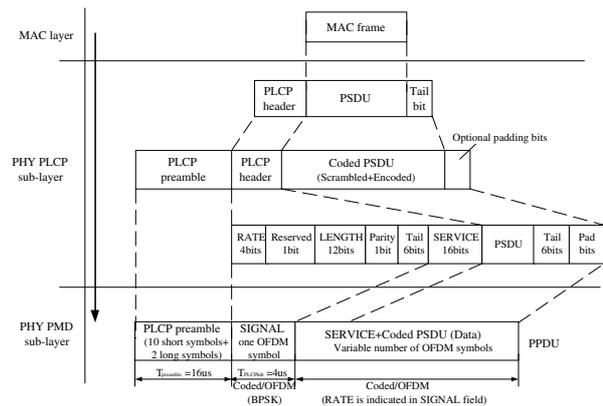
Figure 3: PHY frame formats (PPDU, PSDU) for 802.11a OFDM

4   Analytical model

In this section we present an analytical model in both congested and error-prone channels. In this work, we do not consider the possible capture effects in which the receiver with the strongest receiving power could capture

---

[3]The mechanism used to select dynamically one out of the multiple available data transmission rates at a given time is referred to as a link adaptation scheme.

[4]The optional padding bits are used to make the resulting bit string into a multiple of OFDM symbols.

its sending signal: in our case a *collision* occurs when more than one STA start transmissions simultaneously. For a tagged station, the probability of a collision seen by a packet being transmitted on the channel is denoted by p_c. It is a station-dependent probability. The probability that two or more STAs transmit packets in the same slot is denote by P_C . On the other hand, a transmission error event is independent of a collision event. In case the two events occur simultaneously, we treat them as a single collision event. A transmission *error* event is counted only if one STA transmits and its packet is corrupted due to transmission errors. We denote by p_e the frame error probability. P_E is defined as the probability that there is a transmission error on the channel for an un-collided transmitted frame. We assume that the wireless channel is a Gaussian channel, in which each bit has the same bit error probability, and bit errors are identically and independently distributed (i.i.d.) over the whole frame. Although the Gaussian channel model cannot capture the multi-path fading characteristics of a wireless channel, it is widely used due to its simplicity. We ignore the effects of distance in which different STAs can have different bit error probabilities and different frame error probabilities. In summary, we make the following assumptions:

- Fixed number of STAs with saturated traffic sources, i.e., each STA has always frames available for transmission.

- No hidden terminals [14] and no capture effects [15].

- No link-adaptation mechanism: each STA chooses a static transmission mode and a fixed PHY data rate.

- A Gaussian wireless channel.

In our model all the STAs are assumed to perform the same backoff behavior. It is called a *homogenous case*.

Hence, the following analysis is divided into two parts: First, we investigate the backoff behavior of a single STA with a Markov chain model in congested and error-prone channels. We compute the stationary probability τ that the STA transmits a data frame in a random chosen time slot. Second, by analyzing the events that occur within a randomly chosen time slot, we obtain the system saturation throughput as a function of τ.

Table 1 recapitulates the notations used in this paper.

4.1 Markov chain for a single STA

Similar to [2] and [4], we use a discrete-time Markov chain model to study the random backoff behavior of a STA. The key difference between our model and the Markov chain models in [2] and [4] is that we introduce a new probability, p_f, as shown in Figure 4. p_f stands for the frame failure transmission probability. Either a collision event or a transmission error event results in a failure transmission and thus an increase of the CW size. We also assume that at each transmission attempt, regardless of the number of retransmissions, each frame has a constant and independent failure probability p_f. Thus the transmission failure probability of the STA can be expressed as:

$$p_f = 1 - (1 - p_c)(1 - p_e) = p_c + p_e - p_e p_c, \quad (1)$$

Here, p_e stands for the frame error rate (FER) of a data frame or an ACK frame. We can obtain it by assuming that the two events "data frame corrupted" and "ACK frame corrupted" are independent:

$$p_e = p_e^{data} + p_e^{ack} - p_e^{data} p_e^{ack}, \quad (2)$$

where p_e^{data} and p_e^{ack} are FERs of data frames and ACK

6

n	Number of STAs	L_{Pld}	MAC payload size (0-2304 bytes)
W_0	Minimum contention window size	L_{MAChdr}	MAC header size, including 32 bit CRC
$W_{m'}$	Maximum contention window size	L_{data}	MAC data frame size: $L_{data} = L_{MAChdr} + L_{Pld}$
N_{BpS}	Number of encoded bits per OFDM symbol	L_{ack}	MAC layer ACK frame size
T_{SIFS}	Duration of SIFS	L_{SER}	802.11a PHY layer SERVICE fields size (16 bits)
T_{DIFS}	Duration of DIFS	L_{TAIL}	802.11a PHY layer TAIL fields size (6 bits)
T_{EIFS}	Duration of EIFS	R	PHY layer data rate
T_{Pld}	Duration to transmit a MAC data payload	S	MAC layer saturation throughput
T_{data}	Duration to transmit a MAC data frame	p_c	Collision probability seen by a transmitted packet
T_{ack}	Duration to transmit an ACK frame	p_e	Frame error probability (FER)
$T_{PLCPpreamble}$	Duration of a PLCP preamble	p_b	Bit error rate (BER)
$T_{PLCPhdr}$	Duration of a PLCP header	P_I	Probability of no transmission in a slot
T_{PHYhdr}	Sum of $T_{PLCPpreamble}$ and $T_{PLCPhdr}$	P_E	Probability of transmission errors in a slot
T_{symbol}	Interval of an OFDM symbol in 802.11a	P_C	Probability of collisions in a slot
p_f	Transmission failure probability of a STA	P_S	Probability of successful transmissions in a slot
δ	Propagation delay	E_b	Energy level per transmitted bit
σ	PHY layer actual time slot	N_0	Noise level

Table 1: Notations
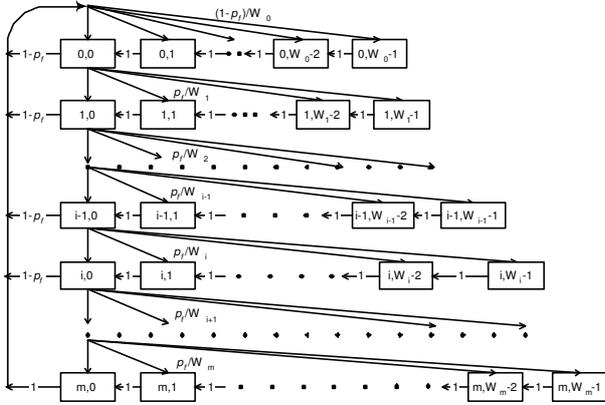
Figure 4: Markov chain model for backoff window size in error-prone networks

frames respectively.

Assuming the bit errors are uniformly distributed over the whole frame, the FER of a data frame, p_e^{data}, can be calculated as:

$$p_e^{data} = 1 - (1 - p_b)^{L_{data}}. \tag{3}$$

Similarly, we obtain the FER of an ACK frame, p_e^{ack}:

$$p_e^{ack} = 1 - (1 - p_b)^{L_{ack}}. \tag{4}$$

The bit error rate, p_b, can be estimated by measuring the bit-energy-to-noise ratio for different modulation schemes: For both BPSK and QPSK modulations, p_b can

be calculated by [16]:

$$p_b = Q(\sqrt{2\frac{E_b}{N_0}}), \tag{5}$$

and for M-ray QAM (M can be 16 or 64 in 802.11a), p_b can be obtained from the following formula [16]:

$$p_b \approx 4(1 - \frac{1}{\sqrt{M}})Q(\sqrt{\frac{3E_b}{(M-1)N_0}}), \tag{6}$$

where $\frac{E_b}{N_0}$ is the bit-energy-to-noise ratio of the received signal and *Q-function* is defined as:

$$Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{\frac{-t^2}{2}} dt. \tag{7}$$

Let b(t) and s(t) be the stochastic processes representing the *backoff_time* and the backoff stage respectively for a given STA at time slot t. The bidirectional process {b(t), s(t)} can be modelled by a Markov chain as shown in Figure 4. Let $b_{i,k} = \lim_{t\to\infty} P\{s(t) = i, b(t) = k\}$ be the stationary distribution of the Markov chain, where $i \in [0, m]$, $k \in [0, W_i - 1]$. The maximum CW size can be calculated by $W_{m'} = 2^{m'} W_0$, where m' represents the number for doubling the CW size up to the maximal CW size. m denotes the maximum number of retransmissions. m can be larger than, equal to, or less than m' [1]. As shown in Figure 4, if $m \geq m'$, a corrupted frame

can be retransmitted several times until reaching its retry limit. Meanwhile, the CW size for this STA can be increased up to its maximum size, $W_{m'}$. On the other hand, if $m < m'$, before the CW size reaches $W_{m'}$, a frame has to be dropped when the number of its retransmission reaches the retry limit. Hence, we have the following

$$W_i = \begin{cases} 2^i W_0, & i \le m' \\ \\ W_{m'} = 2^{m'} W_0, & i > m'. \end{cases} \quad (8)$$

As specified in [1], for frames with length less than the $RTSThreshold$, the default value of m is 7 ($ShortRetryLimit$), whereas for frames longer than the $RTSThreshold$, it is set to 4 ($LongRetryLimit$).

In our Markov chain, the only non-null one-step transition probabilities[5] are:

$$\begin{cases} P\{i,k|i,k+1\} = 1, & 0 \le i \le m, 0 \le k \le W_i - 2 \\ \\ P\{0,k|i,0\} = (1-p_f)/W_0, & 0 \le i \le m-1, 0 \le k \le W_0 - 1 \\ \\ P\{i,k|i-1,0\} = p_f/W_i, & 1 \le i \le m, 0 \le k \le W_i - 1 \\ \\ P\{0,k|m,0\} = 1/W_0, & 0 \le k \le W_0 - 1. \end{cases}$$

The above four transition probabilities account respectively for: 1) the decrements of the backoff_time when the channel is sensed idle for a time slot; 2) after a successful transmission, the backoff_time of the new frame starts from the backoff stage 0; 3) a failure transmission (either due to a collision or an error) leads to the increase of backoff stages; 4) at the maximum backoff stage (i.e., the m-th stage), the CW size will always be reset. This considers the two cases either the transmission is unsuc-

---

[5]Similar to [2], the transition probabilities are expressed in the short notation: $P\{i,k|j,l\} = P\{s(t+1) = i, b(t+1) = k|s(t) = j, b(t) = l\}$.

cessful but it reaches the retry limit, or the transmission is successful, and the CW size should also be reset.

Let $b_{i,k}$ be the stationary distribution of the Markov chain. First, we have:

$$b_{i,0} = p_f^i b_{0,0}, \quad 0 \le i \le m. \quad (9)$$

Owing to the chain regularities, for each $k \in (0, W_i - 1)$, we have:

$$b_{i,k} = \frac{W_i - k}{W_i} \begin{cases} (1-p_f) \sum_{j=0}^{m-1} b_{j,0} + b_{m,0}, & i = 0, \\ \\ p_f b_{i-1,0}, & 0 < i \le m. \end{cases} \quad (10)$$

Using (9), Equation (10) can be simplified as:

$$b_{i,k} = \frac{W_i - k}{W_i} b_{i,0}, \quad 0 \le i \le m. \quad (11)$$

Therefore, with (9), (11), and (8), $b_{0,0}$ is obtained through the following normalization condition:

$$1 = \sum_{i=0}^{m} \sum_{k=0}^{W_i - 1} b_{i,k} = \sum_{i=0}^{m} p_f^i b_{0,0} \frac{W_i + 1}{2}, \quad (12)$$

from which we get:

$$b_{0,0} = \begin{cases} \frac{2(1-2p_f)(1-p_f)}{(1-p_f)W(1-(2p_f)^{m+1})+(1-2p_f)(1-p_f^{m+1})}, & m \le m', \\ \\ \frac{2(1-2p_f)(1-p_f)}{Z}, & m > m', \end{cases}$$

where

$$Z = (1-p_f)W(1-(2p_f)^{m'+1}) + (1-2p_f)(1-p_f^{m+1}) + W2^{m'} p_f^{m'+1}(1-2p_f)(1-p_f^{m-m'}).$$

From the Markov chain in Figure 4, we can now calculate the probability τ that a STA transmits in a ran-

domly chosen time slot. Since any transmission occurs only when its backoff_time reaches zero, τ can be expressed as:

$$\tau = \sum_{i=0}^{m} b_{i,0}$$

$$= \begin{cases} \dfrac{2(1-2p_f)(1-p_f^{m+1})}{(1-p_f)W(1-(2p_f)^{m+1})+(1-2p_f)(1-p_f^{m+1})}, & m \leq m', \\[3mm] \dfrac{2(1-2p_f)(1-p_f^{m+1})}{Z}, & m > m'. \end{cases} \quad (13)$$

Assuming that each frame collides with a constant and independent probability p_c, we have the following relation:

$$p_c = 1 - (1-\tau)^{n-1}. \quad (14)$$

Combining Equations (1) and (14), we get the expression of p_f:

$$p_f = 1 - (1-p_e)(1-\tau)^{n-1}, \quad (15)$$

where p_e is obtained from Equation (2).

Equations (13) and (15) represent a nonlinear system with two unknown variables, τ and p_f, which can be solved numerically.

4.2 System saturation throughput

The saturation throughput S is defined as successfully transmitted payloads in a randomly chosen time slot duration:

$$S = \frac{E[L_{Pld}]}{E[T]}, \quad (16)$$

where $E[L_{Pld}]$ is the expected value of the successfully transmitted payload sizes, and $E[T]$ denotes the corresponding expected value of time slot durations.

To calculate S, we first analyze a randomly chosen time slot. As shown in Figure 5, there are five kinds of possible time slot durations: 1) T_I, the idle slot duration; 2) T_S, the average time the channel is sensed busy because of a successful transmission; 3) T_C, the average time the channel is sensed busy because of a collision. Since hidden terminal is not considered in this paper, only the data frames can get collided, and there are no collisions for ACK frames; 4) T_E^{data}, the average time the channel is sensed busy because of a transmission error for a data frame; 5) T_E^{ack}, the average time the channel is sensed busy because of a transmission error for an ACK frame.
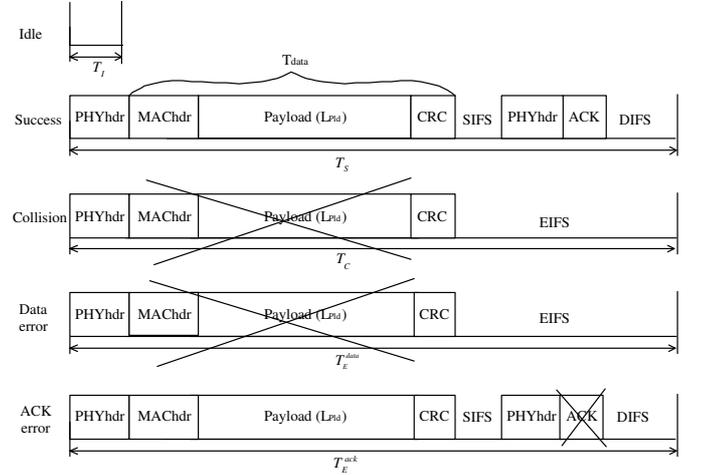

Figure 5: Time slot durations under an error-prone channel

The first three slots (T_I, T_S, T_C) have been studied in the literature (e.g. [2]-[7]). However, the calculation of T_C in the literature ($T_C = T_{PHYhdr} + T_{data} + \delta + T_{DIFS}$) is not in accordance with the 802.11 standard [1]. As shown in Figure 1, the channel should be reserved for an EIFS interval (T_{EIFS}) after a data frame collision is detected:

$$T_{EIFS} = T_{SIFS} + T_{PHYhdr} + T_{ack} + \delta + T_{DIFS}. \quad (17)$$

Given that the 802.11 MAC protocol cannot differentiate between a collision and a transmission error, we have $T_E^{data} = T_C$. When there is a transmission failure (either due to a collision or a transmission error), other STAs than the transmitter use an EIFS interval as the waiting period, and the transmitter waits for $T_{ACKOut} + T_{DIFS}$ before possible next transmission. We then have:

$$T_{ACKOut} = T_{SIFS} + T_{PHYhdr} + T_{ack} + \delta. \quad (18)$$

Hence, we conclude that in the case of a failure data frame transmission, all the STAs wait for the same duration before the channel is idle again and the channel is sensed busy for the duration of the data frame transmission time plus an EIFS interval, see T_C and T_E^{data} in Figure 5.

As explained in Section 2, T_E^{ack} is different from T_E^{data}. If a data frame is successfully received but only its ACK is corrupted, all the other STAs will treat this event as a successful transmission and reserve the channel for $NAV + DIFS$, thus we obtain $T_E^{ack} = T_S$.

If no STA transmits during a time slot, i.e., the time slot is idle, then the n STAs wait for the shortest time slot T_I. Otherwise, the duration can be expressed by the summation of the time that the channel is sensed busy and the time the system waits after the channel becomes idle again. For example, T_S is the sum of a DIFS interval and the successful transmission time durations of a data frame followed by an ACK frame. T_C, T_E^{data}, and T_E^{ack} refer respectively to the interval that the channel is occupied because of collisions, data frame errors and ACK frame errors. Thus, the five different time slots are as follows:

$$
\begin{cases}
T_I = \sigma \\
T_S = 2T_{PHYhdr} + T_{data} + 2\delta + T_{SIFS} + T_{ack} + T_{DIFS} \\
T_C = T_{PHYhdr} + T_{data} + \delta + T_{EIFS} \\
T_E^{data} = T_{PHYhdr} + T_{data} + \delta + T_{EIFS} \\
T_E^{ack} = T_S,
\end{cases}
$$
$$(19)$$

where T_{data} and T_{ack} denote the transmission time of a data frame and an ACK frame respectively. They are PHY-layer dependent. Based on the frame transmission analysis done in Section 3, we obtain:

$$T_{data} = T_{symbol} \, Ceiling(\frac{L_{SER} + L_{TAIL} + L_{data}}{N_{BpS}}), \quad (20)$$

$$T_{ack} = T_{symbol} \, Ceiling(\frac{L_{SER} + L_{TAIL} + L_{ack}}{N_{BpS}}), \quad (21)$$

where *Ceiling*() is a function that returns the smallest integer value greater than or equal to its argument value. L_{SER} denotes the size of the SERVICE field and L_{TAIL} denotes the TAIL field in 802.11a. They are all listed in Table 1. These formulas consider that the PHY layer frame transmission is in unit of OFDM symbols.

Now, we analyze the corresponding probabilities for a random slot duration. Let P_I be the probability that no transmission occurs in a time slot, it is expressed as:

$$P_I = (1 - \tau)^n, \quad (22)$$

where τ represents the probability that a STA starts transmission in a randomly chosen slot.

P_E^{data} stands for the probability that a transmission error occurs on a data frame in a time slot; this occurs when one and only one STA transmits in a time slot and the data

frame is corrupted because of transmission errors. P_E^{ack} denotes the probability that a data frame transmission is successful but the corresponding ACK frame is corrupted due to transmission errors. A transmission error in the data frame or in the ACK frame can be detected by one of the following techniques: 1) A CRC frame check failure. 2) The PHY layer sends an error signal to the MAC layer when the receiving frame cannot be decoded or when the incoming signal is lost in the middle of a frame reception.

Thus, P_E^{data} and P_E^{ack} can be expressed using the following two equations:

$$P_E^{data} = n\tau(1-\tau)^{n-1}p_e^{data}, \qquad (23)$$

$$P_E^{ack} = n\tau(1-\tau)^{n-1}(1-p_e^{data})p_e^{ack}, \qquad (24)$$

The probability P_S for a successful transmission in a slot is obtained only when one STA transmits a frame and there is no error neither in the data frame nor in the ACK frame.

$$P_S = n\tau(1-\tau)^{n-1}(1-p_e^{data})(1-p_e^{ack}). \qquad (25)$$

The probability that there is a collision on a time slot is equal to:

$$P_C = 1 - (1-\tau)^n - n\tau(1-\tau)^{n-1}. \qquad (26)$$

Finally, the system saturation throughput can be computed as follows:

$$S = \frac{P_S L_{Pld}}{T_I P_I + T_S P_S + T_E^{data} P_E^{data} + T_E^{ack} P_E^{ack} + T_C P_C}, \qquad (27)$$

where T_I, T_S, T_E^{data}, T_E^{ack} and T_C can be obtained from Equation (19). P_I, P_E^{data}, P_E^{ack}, P_S and P_C can be calculated from Equations (22)–(26) respectively.

5 Model validation

We have validated our analytical model using the network simulation tool, *NS-2* [17]. We have made the following modifications to the NS simulation environment:

1) We set that the transmission time value for the 802.11a PLCP preamble and PLCP header to 20 μs as specified in the 802.11a standard [13]. 2) Static routing is used between nodes in order to study the performance of the pure MAC layer protocol. 3) Transmission errors are generated according to the Gaussian channel assumption.

As an example, we choose the 802.11a PHY layer with a data rate equal to 6 Mbps (model-1, BPSK modulation) in the following simulations. The transmitting power used for each STA is assumed to be high enough to cover a 250 meters transmission range. The distance between two neighbors is 1 meter. In this way, every STA is able to listen each other and thus there are no hidden terminals in the system. Table 2 summarizes the MAC and PHY layers' parameters used in the simulations.

T_{SIFS} (μs)	16	L_{MAChdr} ($bits$)	224
σ (μs)	9	L_{ack} ($bits$)	112
T_{DIFS} (μs)	34	δ (μs)	1
T_{PHYhdr} (μs)	20	T_{symbol} (μs)	4
W_0	16	N_{BpS}	24

Table 2: The MAC and PHY parameters for 802.11a

To validate our model, we compare the simulation results with those analytical results obtained from the Bianchi's model [2], the model in [4] and our model. For this simulation, the wireless channel BER value is 10^{-5} and the number of STAs varies from 5 to 80 to investigate the impact of varying channel conditions with transmission errors and congestions. For a given number of STAs, we run 3 simulations with different random seeds. Each symbol represents a simulation result. Some symbols are superposed because those simulation results are

11

very close to each other. As shown in Figure 6, our analytical model matches the simulation results much closer than the models in [2] and [4]. Their models overestimate the saturation throughput of 802.11 because they do not consider channel errors in the Markov chain model.
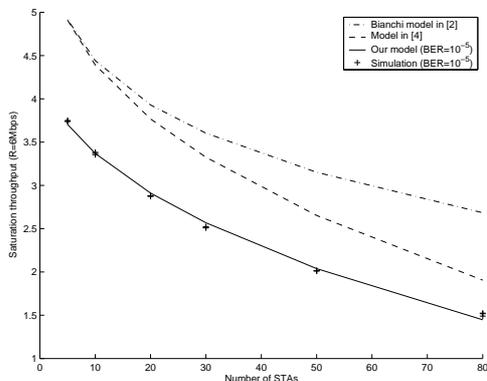

Figure 6: Model validation: comparison with simulations and other models (6Mbps 802.11a, $BER = 10^{-5}$)

In Figure 7, we study the impact of the frame size according to the channel conditions. From the figure we can see that our analytical model is very accurate if the channel BERs are not very high (i.e. when $BER = 10^{-6}$ and $BER = 10^{-5}$), and slightly overestimates the saturation throughput on a very noisy channel (i.e. when $BER = 10^{-4}$). Even for the latter case, the difference between the model and the simulation is less than 1%, which means that our model is precise enough to predict the performance of an 802.11 system in error-prone channels. An interesting observation from this figure is that a larger frame size results in a higher throughput when the channel BER is very low, which means a large frame size can significantly improve the data throughput when channel is in a good condition. However, when the channel is in a bad condition, increasing the frame size degrades the throughput, and thus, the optimal frame size should depend on the channel conditions.

As a first application of our model, we have computed the optimal frame size according to the channel condi-
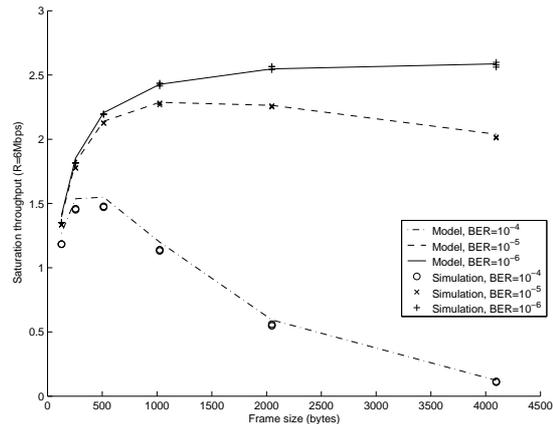

Figure 7: Model validation: analysis vs. simulations for 50 STAs (6Mbps 802.11a, different BER values)

tions as shown in Figure 8. Here, the optimal frame size refers to the payload size at the MAC layer which gives the maximum saturation throughput for a given channel condition (BER) and a certain number of STAs (n). We increase the data payload size from 128 bytes to 4500 bytes with a step of 128 bytes to find the optimal frame size which provides the maximum saturation throughput. As expected, under a saturated condition, the optimal frame size decreases when the channel BER value increases, and it almost has no relation with the number of STAs. It is because packets are always available in all the cases and thus only the channel BER value affects the optimal frame size. In Figure 8 there is a discrepancy at $BER = 2 \times 10^{-5}$ between the optimal frame size for 80 STAs and for smaller number of STAs. Note that this point is only two steps of 128 bytes below the other points with different number of users. This is due to the fact that the saturation throughput curve is too flat in the area close to the maximum point and thus our algorithm underestimates slightly the optimal frame size for 80 STAs than for other number of STAs.

Figure 9 shows the saturation throughput performance obtained with the model for different numbers of STAs according to the channel conditions. As expected, if the

channel is not very noisy (i.e., when $BER < 10^{-5}$), the saturation throughput decreases when the number of STAs increases. However, if the channel is very noisy (e.g., when $BER = 10^{-3}$), the saturation throughput increases when the number of STAs increases. Actually in the latter case, the bulk of frames are dropped due to transmission errors. The more STAs join, the more likely data frames are successfully transmitted through the channel and thus the higher overall throughput is achieved.
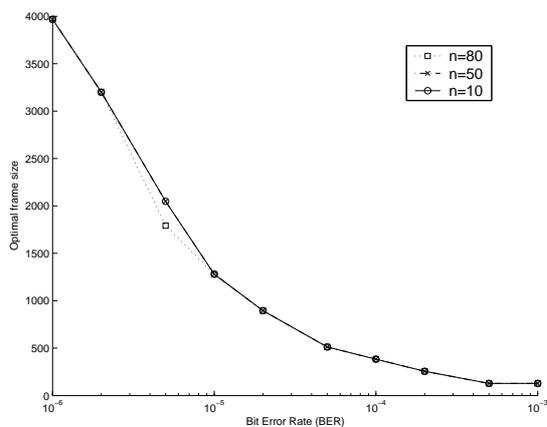

Figure 8: Optimal payload sizes at the MAC layer according to the channel conditions
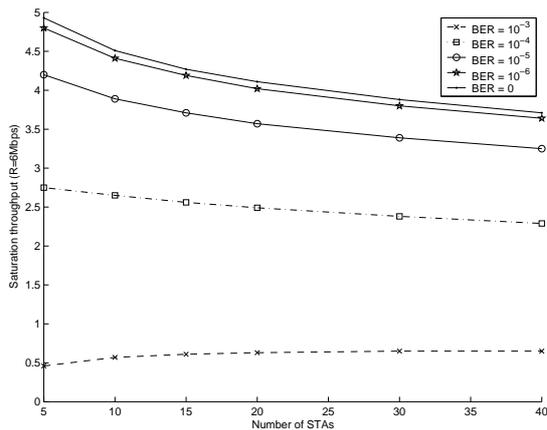
Figure 9: Saturation throughput for different numbers of STAs according to the channel conditions

6   Conclusion

In this paper, we have presented an analytical model to compute the 802.11 MAC layer saturation throughput for the DCF mechanism in both congested and error-prone wireless channels. Simulation results show that our model is very accurate. As a first application of our model, we have computed the optimal payload sizes according to the channel conditions. Our results confirm that transmission errors have a significant impact on the 802.11 MAC layer throughput performance. Future work includes delay analysis in congested and error channels.

References

[1] IEEE 802.11 WG. International standard for information technology -local and metropolitan area networks, part 11: wireless LAN MAC and PHY specifications, 1999.

[2] Bianchi G. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications* 2000; **18**(3): 535-548.

[3] Vishnevsky V, Lyakhov A. IEEE 802.11 LANs: saturation throughput analysis with seizing effect consideration. *Journal of Cluster Computing* 2002; **5**: 133-144.

[4] Wu H, Peng Y, Long K, Cheng S, Ma J. Performance of reliable transport protocol over IEEE 802.11 wireless LAN: analysis and enhancement. *Proceedings of IEEE INFOCOM* 2002; **2**: 599-607.

[5] Cantieni G, Ni Q, Barakat C, Turletti T. Performance analysis under finite load and improvements for multirate 802.11. *Elsevier Computer Communications Journal* 2005; **28**(10): 1095-1109. DOI: 10.1016/j.comcom.2004.07.023.

[6] Chatzimisios P, Boucouvalas AC, Vitsas V. Performance analysis of IEEE 802.11 DCF in presence of transmission errors. *Proceedings of IEEE ICC* 2004; **7**: 3854-3858.

[7] Velkov ZH, Spasenovski B. Saturation throughput - delay analysis of IEEE 802.11 DCF in fading channel. *Proceedings of IEEE ICC* 2003; **1**: 121-126.

[8] Yin J, Wang X, Agrawal DP. Optimal packet size in error-prone channel for IEEE 802.11 distributed coordination function. *Proceedings of IEEE WCNC* 2004; **3**: 1654-1659.

[9] He JH, Tang ZY, Yang ZK, et al. Performance evaluation of distributed access scheme in error-prone channel. *Proceedings of IEEE TENCON* 2002; **2**: 1142-1145.

[10] Vishnevsky V, Lyakhov A. 802.11 LANs: saturation throughput in the presence of noise. *Proceedings of IFIP Networking* 2002; **2345**: 1008-1019.

[11] Yeo J, Agrawala A. Packet error model for the IEEE 802.11 MAC protocol. *Proceedings of IEEE PIMRC* 2003; **2**: 1722-1726.

[12] Tay YC, Chua KC. A capacity analysis for the IEEE 802.11 MAC protocol. *Wireless Networks* 2001; **7**(2): 159-171.

[13] IEEE 802.11 WG. Part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications: high-speed physical layer in the 5 GHz band, IEEE 802.11a, 1999.

[14] Huang KC, Chen KC. Interference analysis of nonpersistent CSMA with hidden terminals in multicell wireless data networks. *Proceedings of IEEE PIMRC* 1995; **2**: 907-911.

[15] Kim JH, Lee JK. Capture effects of wireless CSMA/CA protocols in rayleigh and shadow fading channels. *IEEE Transactions on Vehicular Technology* 1999; **48**4: 1277-1286.

[16] Rappaport TS. *Wireless Communications: Principles and Practice*. Prentice Hall: New Jersey, USA, 1996.

[17] NS-2 simulator, URL http://www.isi.edu/nsnam/ns/ [Date of access: April 2004].

Tianji Li received the B.S. and M.S. degrees in computer science from JiLin and ZhongShan Universities, China, in 1998 and 2001, respectively, and the M.S. degree in networking and distributed computation from the University of Nice Sophia Antipolis, France, in 2004. Currently, he is working towards the Ph.D. degree at the Hamilton Institute, National University of Ireland Maynooth. From 2001 to 2003, he was a software engineer at the Beijing Research Institute of Huawei Technologies, China. His research interests are performance evaluation and optimization in wireless networks. (E-mail: tianji.li@nuim.ie)

Thierry Turletti received the M.S. (1990) and the Ph.D. (1995) degrees in computer science both from the University of Nice Sophia Antipolis, France. He did his PhD studies in the RODEO group at INRIA Sophia Antipolis. During the year 1995-96, he was a postdoctoral fellow in the Telemedia, Networks and Systems group at LCS, MIT. He is currently a research scientist at the Planète group at INRIA Sophia Antipolis. His research interests include multimedia applications, congestion control and wireless networking. Dr. Turletti currently serves on the editorial board of Wiley Journal of Wireless Communications and Mobile Computing. (E-mail: Thierry.Turletti@sophia.inria.fr)

Qiang Ni received the BEng., M.S. and Ph.D. degrees from Hua Zhong University of Science and Technology (HUST), China in 1993, 1996 and 1999 respectively. From 1999 to 2001, he was a post-doctoral fellow in the wireless communication laboratory, HUST. He visited the Microsoft Research Asia Lab during year 2000. In 2001, he joined INRIA, France, where he was a research staff member at the Planete group. He is currently a senior researcher at the Hamilton Institute, National University of Ireland Maynooth. Since 2002, he has been a voting member for the IEEE 802.11 working group. He serves as a TPC member for several communication conferences such as GlobeCom 05, WirelessCom 05, WiOPT 05, etc. His research interests include wireless network protocol design and performance analysis. He is a member of IEEE. (E-mail: Qiang.Ni@ieee.org)

Yang Xiao is an assistant professor of Department of Computer Science at the University of Memphis. He is an IEEE Senior member. He was a voting member of IEEE 802.11 Working Group from 2001 to 2004. He currently serves as an associate editor or on editorial boards for six refereed journals including (Wiley) International Journal of Communication Systems, (Wiley) Wireless Communications and Mobile Computing, EURASIP Journal on Wireless Communications and Networking, International Journal of Wireless and Mobile Computing, etc. He serves as a (lead/sole) guest editor for four journal special issues, an editor for four books, and a NSF panelist. His research areas include wireless networks and mobile computing. (E-mail: yangxiao@ieee.org)