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 LANs. Much work has been done on 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. 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 the predominant technology for wireless access in local areas: the 802.11b WiFi networks with physical (PHY) layer data rates up to 11 Mbps in the 2.4 GHz frequency band have been widely deployed in hotspots and offices. Furthermore, 802.11a in the 5 GHz band, and 802.11g in dual bands (2.4 GHz and 5 GHz), are being deployed to provide PHY layer data rates up to 54 Mbps. To further increase data rates and throughput, the 802.11 working group created a new task group, namely 802.11n, which focuses on the standardization issues of next-generation WLANs to achieve 100 Mbps net throughput.

In the 802.11 protocol stack, the medium access control (MAC) layer plays a key role in determining the channel efficiency and quality-of-service (QoS) for upper-layer applications. 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 (point coordination function, PCF, and hybrid coordination function, HCF), were also proposed by the 802.11/802.11e groups. The latter three mechanisms are based on DCF and required to be compatible with it. Thus, a thorough understanding of the DCF performance in various 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 carried out to model the behavior of DCF with saturated loads in an error-free channel condition (e.g., [2]–[4]).

*Correspondence to: Qiang Ni, Hamilton Institute, National University of Ireland Maynooth, Co. Kildare, Ireland, Tel: +353-1-7086463, Fax: +353-1-7086269. Email: Qiang.Ni@ieee.org

†The work of Qiang Ni and Tianji Li was supported in part by the Science Foundation Ireland under Grant 03/IN3/I396. Part of this work was done while they were with INRIA, France.

Among them, Bianchi's two-dimensional Markov chain model [2] is a fundamental one. In [3], the seizing effect is added into Bianchi's model, considering that the station which just finished a successful transmission has a better chance to access and seize the channel than the others. Another extension is provided in [4] by considering the 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 error channel is assumed in which a constant channel bit error rate (BER) is supposed to be known in advance. The channel BER is then introduced into Bianchi's model[1]. To the best of our knowledge, most of the existing work only considers the error probability of data frames. [10] is the only one which also 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. Furthermore, in [6]–[8], the impact of EIFS interval has not been modeled when a transmission failure occurs. [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 802.11 PHY layers (e.g., 802.11a, 802.11b, 802.11g and the 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 and simulations.

The remainder of this paper is organized as follows. Section 2 explains how the 802.11 DCF protocols handle frame collisions and transmission errors. 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

In this section, we first explain the principles of the DCF protocol. We then present how DCF handles collisions and frame transmission errors.

The basic access mechanism of DCF is a *carrier sense multiple access with collision avoidance* (CSMA/CA) scheme with a binary exponential backoff. In DCF, there are two kinds of carrier sensing mechanisms: the PHY layer carrier sensing and the MAC layer virtual carrier sensing. The PHY layer carrier sensing discovers whether the wireless medium is busy or not through a clear channel assessment function [1]. Then, the PHY layer sends an indication of carrier status to the MAC layer. On the other hand, the virtual carrier sensing is

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

optionally used by a transmitting station (STA) as follows: Before transmitting a frame, a sender fills in the duration field of the frame's MAC header a value which indicates how long it expects to use the medium. Then, other STAs hearing its transmission can update their local network allocation vectors (NAVs) to this time duration and defter their transmissions until the NAV timers count down to zero at a uniform rate. When a NAV timer reaches zero, it indicates that the medium is virtually idle. By combining the virtual carrier sensing with the PHY carrier sensing, DCF implements both CSMA and CA, and it works as follows: If one STA intends to transmit a frame and the medium has been sensed idle for an interval of time equal to a distributed interframe space (DIFS), the STA must defer its transmission for a random interval called *backoff_time*. The *backoff_time* is a random number generated uniformly from the interval [0, CW-1], where CW is called the contention window size. It is doubled after each unsuccessful transmission until reaching a maximum value called CW_{max}, or it is reset to a minimum value called CW_{min} after a successful transmission or if the frame is dropped. The backoff time is slotted and a station is allowed to transmit only at the beginning of each slot. The *backoff_time* is decremented by one each time when the medium is sensed idle for one slot time, otherwise it is frozen. 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, or transmission errors appear if the channel conditions are poor, which can be caused by channel fading, path loss, thermal noise, or interferences from other radio sources (e.g., Bluetooth devices, microwave ovens). In order to notify the sender that the frame has been received successfully, a positive acknowledgement (ACK) is required to be sent out by the receiver after receiving the data frame correctly. 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 its transmission was failed (either due to collisions or transmission errors). Then it schedules a retransmission by entering again a backoff process with a double-sized CW value until the maximum *retry limit* is reached.

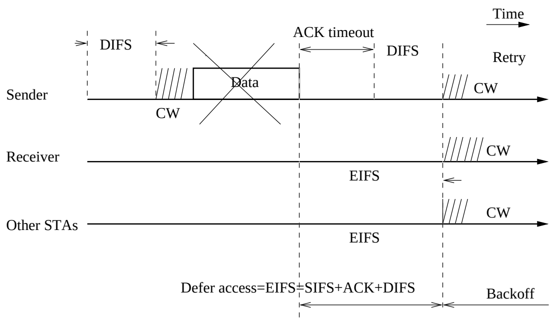

Figure 1: An example of unsuccessful transmission in DCF due to corrupted data

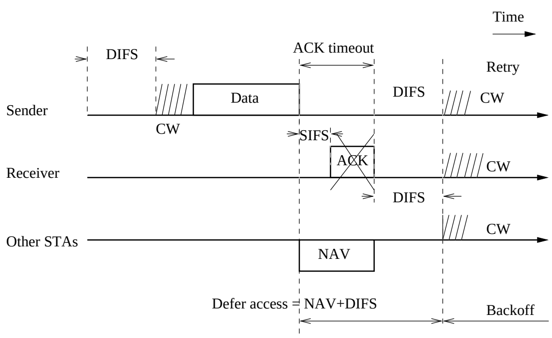

Figure 2: An example of unsuccessful transmission in DCF due to corrupted ACK

It should be pointed out that collisions and transmission errors are not differentiated by the 802.11 MAC protocol. With DCF, if a transmitter does not receive an ACK frame, it increases its CW size and retransmits the frame given the *retry limit* is not reached. Ideally, the CW size should be increased only when frame losses are due to collisions in order to decrease congestion, since increasing the CW size in the case of transmission errors may degrade the data throughput and may increase the transmission delays. However, the 802.11 standard cannot distinguish collisions from transmission errors at the MAC layer and it handles them in a same way. 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 is a 32 bit cyclic redundancy code (CRC). As shown in Figure 1, an EIFS interval[2] is the sum of DIFS plus SIFS and plus 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 CTS frame (14 bytes) if it is ready to receive. Once the transmitter receives the CTS frame, it transmits the data frame. 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.

Two retry counters associated with each MAC frame are set: a short and a long retry counter. For a frame whose length is less than or equal to the *RTSThreshold*, the short retry counter is used. 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 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 adopts the orthogonal frequency division multiplexing (OFDM) technology [13]. The basic idea of the OFDM technology is to divide a high-speed binary signal into a number of parallel low rate bit streams and modulating each of these data streams onto individual sub-carriers. In 802.11a, 52 sub-carriers are

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

introduced, of which 48 sub-carriers carry actual data and 4 sub-carriers are pilots that facilitate phase tracking for coherent demodulation. Each low rate bit stream is used to modulate a subcarrier from one of the channels in the 5 GHz band. A modulation operation involves translating a data stream into a sequence of symbols. Each symbol may encode a certain number of bits. The number depends 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

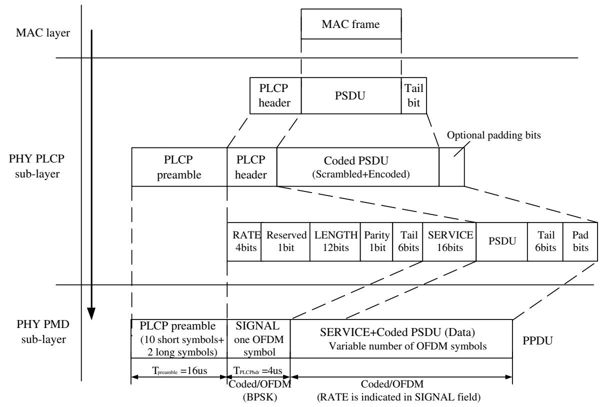

Figure 3: PHY frame formats (PPDU, PSDU) for 802.11a OFDM

[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.

the data rate specified in the RATE field.

4 Analytical model

In this section we present an analytical model for the 802.11 DCF protocol in congested and error-prone channels. In this work, collision and transmission error are considered as two independent events. A *collision* occurs definitely when multiple STAs start transmissions simultaneously[5]. On the other hand, a transmission *error* is considered only if a frame is corrupted due to channel noises. In case both events occur simultaneously, we only treat them as a single collision event.

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.

[5] We do not consider the possible capture effects in which the receiver with the strongest receiving power could capture its sending signal.

Let the number of contending STAs to be a fixed value n. For a given STA, the probability of a collision seen by its packet being transmitted on the channel is denoted by p_c. This is a station-dependent collision probability. On the other hand, the probability that any two or more STAs start transmissions in a same slot is denoted by P_C. The latter one is measured from a system point of view, without referring to any particular STA. Similarly, the frame error probability for a given STA is denoted by p_e. P_E is defined as the probability that there is a transmission error on the channel without looking into any particular STA.

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 any given 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 or a transmission error event results in a failed transmission and thus an increase of the CW size.

Let s(t) and b(t) be the stochastic processes representing the backoff stage and the backoff_time counter

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 one 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

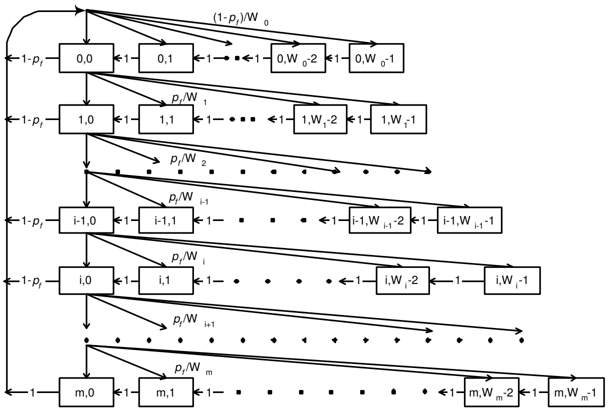

Figure 4: Markov chain model for backoff window size in error-prone networks

respectively for a given STA at time slot t. The bidirectional process {s(t), b(t)} can be modeled 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 $i \in [0, m]$, $k \in [0, W_i - 1]$. m denotes the retry limit number for any transmitted frame. 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 ($LongRetryLimit$). For convenience, CW_{min} is denoted by W_0, and CW_{max} is denoted by $W_{m'}$ which is equal to $2^{m'}W_0$, where m' represents the number of doubling the CW size from CW_{min} to CW_{max}. Note that m' can be larger than, equal to, or less than m [1]. If $m' \leq m$, once the CW reaches $W_{m'}$, it will remain at this value of $W_{m'}$ until it is reset. Accordingly, the frame will be retransmitted until reaching its retry limit. On the other hand, if $m' > m$, before the CW size reaches $W_{m'}$, the frame has to be dropped when the number of its retransmission reaches the retry limit. Hence, we have the following

$$W_i = \begin{cases} 2^i W_0, & i \leq m' \\ \\ 2^{m'} W_0 = W_{m'}, & i > m'. \end{cases} \tag{1}$$

In our Markov chain, the only non-null one-step transition probabilities[6] are:

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

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

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 failed 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 when the transmission is unsuccessful but it reaches the retry limit, or when the transmission is successful.

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

$$b_{i,0} = p_f^i b_{0,0}, \quad 0 \leq i \leq m. \tag{2}$$

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

$$b_{i,k} = \frac{W_i - k}{W_i} \begin{cases} (1-p_f)\sum_{j=0}^{m-1} b_{j,0} + b_{m,0}, & i = 0, \\ \\ p_f b_{i-1,0}, & 0 < i \leq m. \end{cases} \tag{3}$$

Using (2), Equation (3) can be simplified as:

$$b_{i,k} = \frac{W_i - k}{W_i} b_{i,0}, \quad 0 \leq i \leq m. \tag{4}$$

Therefore, with (2), (4), and (1), $b_{0,0}$ is obtained through the following normalization condition:

$$1 = \sum_{i=0}^{m} \sum_{k=0}^{W_i - 1} b_{i,k} = \sum_{i=0}^{m} p_f^i b_{0,0} \frac{W_i + 1}{2}, \tag{5}$$

from which we get:

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

where

$$\begin{aligned} Z =& (1-p_f)W(1-(2p_f)^{m'+1}) + (1-2p_f)(1-p_f^{m+1}) \\ &+ W2^{m'} p_f^{m'+1}(1-2p_f)(1-p_f^{m-m'}). \end{aligned}$$

From the Markov chain in Figure 4, we can now calculate the probability τ that one STA transmits in a randomly chosen time slot. Since any transmission occurs only when its backoff_time reaches zero, τ can be expressed as:

$$\begin{aligned} \tau =& \sum_{i=0}^{m} b_{i,0} \\ =& \begin{cases} \frac{2(1-2p_f)(1-p_f^{m+1})}{(1-p_f)W(1-(2p_f)^{m+1}) + (1-2p_f)(1-p_f^{m+1})}, & m \leq m', \\ \\ \frac{2(1-2p_f)(1-p_f^{m+1})}{Z}, & m > m'. \end{cases} \end{aligned} \tag{6}$$

Note that p_f is still unknown. To calculate p_f, we 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 a given STA can be expressed as:

$$p_f = 1 - (1-p_c)(1-p_e) = p_c + p_e - p_e p_c, \tag{7}$$

Here, p_e stands for the frame error probability (FER) of a MAC data frame or an ACK frame for the given STA. Assuming that the two events "data frame corrupted" and "ACK frame corrupted" are independent, we obtain:

$$p_e = p_e^{data} + p_e^{ack} - p_e^{data} p_e^{ack}, \tag{8}$$

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

Given that the bit errors are uniformly distributed over the whole frame, p_e^{data} and p_e^{ack} can then be calculated as:

$$p_e^{data} = 1 - (1 - p_b)^{L_{data}}, \tag{9}$$

$$p_e^{ack} = 1 - (1 - p_b)^{L_{ack}}, \tag{10}$$

where the bit error rate (p_b) can be estimated by measuring the bit-energy-to-noise ratio: For both BPSK and QPSK modulations, p_b can be calculated by [16]:

$$p_b = Q\left(\sqrt{2\frac{E_b}{N_0}}\right), \tag{11}$$

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

$$p_b \approx 4(1 - \frac{1}{\sqrt{M}})Q\left(\sqrt{\frac{3E_b}{(M-1)N_0}}\right), \tag{12}$$

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

$$Q(x) = \int_x^{\infty} \frac{1}{\sqrt{2\pi}} e^{\frac{-t^2}{2}}\, dt. \tag{13}$$

To calculate p_c, we assume that each frame collides with a constant and independent probability p_c. We then have the following relation:

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

Combining Equations (7) and (14), we get the expression of p_f:

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

where p_e is obtained from Equation (8).

Equations (6) 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 a ratio of successfully transmitted payload size over a randomly chosen time slot duration:

$$S = \frac{E[L_{Pld}]}{E[T]}, \tag{16}$$

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.

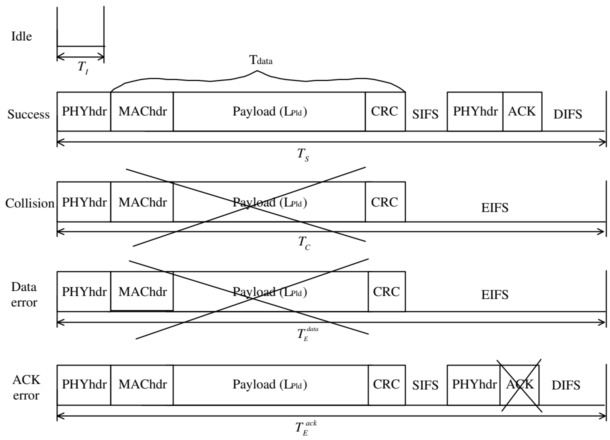

Figure 5: Time slot durations under an error-prone channel

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 duration in which the channel is sensed busy because of a successful transmission; 3) T_C, the duration in which 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 duration that the channel is sensed busy because of a transmission error of a data frame; 5) T_E^{ack}, the duration in which the channel is sensed busy because of a transmission error of an ACK frame.

If there is no transmission during a time slot, i.e., the slot is idle, then the n STAs wait for the shortest slot duration 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 until 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 time interval that the channel is occupied because of collisions, data frame errors and ACK frame errors.

Note that the above three kinds of slots (T_I, T_S, T_C) have been studied before (e.g. [2]–[7]). However, the calculation of T_C in the literature (i.e., $T_C = T_{PHYhdr} + T_{data} + \delta + T_{DIFS}$) is not in accordance with the 802.11 standard [1]. Actually, T_C should be equal to $T_{PHYhdr} + T_{data} + \delta + T_{EIFS}$, and we have

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

Given that the DCF protocol cannot distinguish transmission errors of data frames from collisions, we then obtain $T_E^{data} = T_C$.

More precisely, if there is a failure in a data frame transmission (either a collision or a transmission error), all the STAs except the transmitter should defer their transmissions for a same time interval which is equal to T_C or say T_E^{data}, as shown in Figure 1 and Figure 5. The transmitter should wait for a time duration $T_{PHYhdr} + T_{data} + \delta + T_{ACKOut} + T_{DIFS}$ before starting another transmission, where

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

By a simple calculation, we found this time duration is also equal to T_C. Hence, in the case of a failed data frame transmission, all the STAs defer a same period of time (say T_C or T_E^{data}) before contending the channel again. During this time period, the channel is sensed busy (see Figure 5).

On the other hand, if a data frame was successfully received but its ACK frame was corrupted due to channel noises, other STAs than the communication pair will treat this event as a successful transmission since they can decode the duration field correctly from the transmitted data frame. They sense the channel busy for the time period T_E^{ack}, which is equal to T_S.

In summary, the five different time slots are as follows:

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

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

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

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

where *Ceiling*() is a function that returns the smallest integer 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.

Now, we analyze the corresponding probabilities to have the above slot durations. 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 given STA starts a transmission in a randomly chosen slot.

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.

$$P_S = n\tau(1 - \tau)^{n-1}(1 - p_e^{data})(1 - p_e^{ack}). \quad (23)$$

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

$$P_C = 1 - (1 - \tau)^n - n\tau(1 - \tau)^{n-1}. \quad (24)$$

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:

$$P_E^{data} = n\tau(1 - \tau)^{n-1}p_e^{data}, \quad (25)$$

$$P_E^{ack} = n\tau(1 - \tau)^{n-1}(1 - p_e^{data})p_e^{ack}, \quad (26)$$

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

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

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

5 Model validation

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

1) We set the transmission time of the 802.11a PLCP preamble and PLCP header to 20 μs as specified in the 802.11a standard [13]. 2) Static routing is used 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 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
m	4	m'	6

Table 2: The MAC and PHY parameters for 802.11a

To validate our model, we compare the simulation results with the analytical results obtained from Bianchi's model [2], the model in [4] and our model (see Figure 6). For this simulation, the wireless channel BER value is 10^{-5} and the number of STAs varies from 5 to 80. For a given number of STAs, we run three simulations with different random seeds. Each symbol represents a simulation result. Some symbols are superposed because those simulation results are 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.

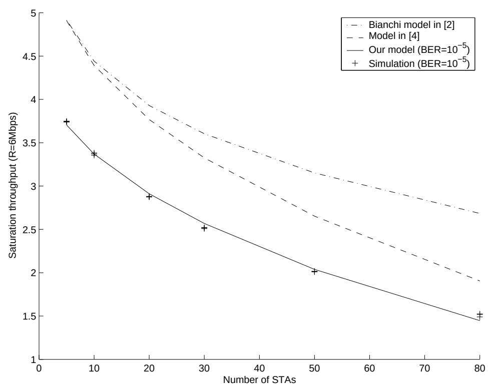

Figure 6: Model validation: comparison with simulations and other models (frame size: 4096 bytes, 6 Mbps 802.11a)

In Figure 7, we study the impact of frame size on the throughput given various channel conditions. We can see that our analytical model is very accurate if the channel BERs are not very high (i.e. $BER = 10^{-6}$ and $BER = 10^{-5}$), and slightly overestimates the saturation throughput on a very noisy channel (i.e. $BER = 10^{-4}$). 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 under a good channel condition. However, when the channel is in a bad condition, large frame size degrades the throughput.

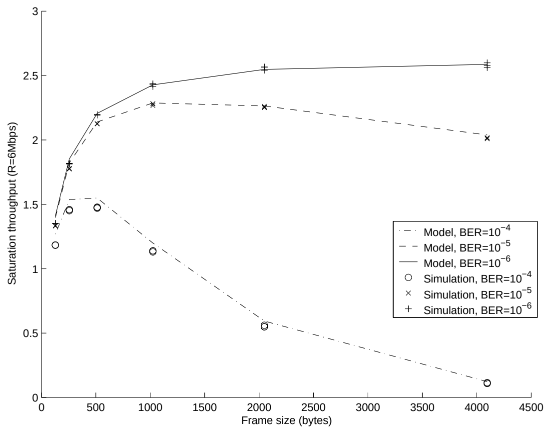

Figure 7: Model validation: analysis vs. simulations for 50 STAs (6 Mbps 802.11a, different BER values)

As a first application of our model, we have computed the optimal frame size according to the channel conditions 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 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 has no relation with the number of STAs.

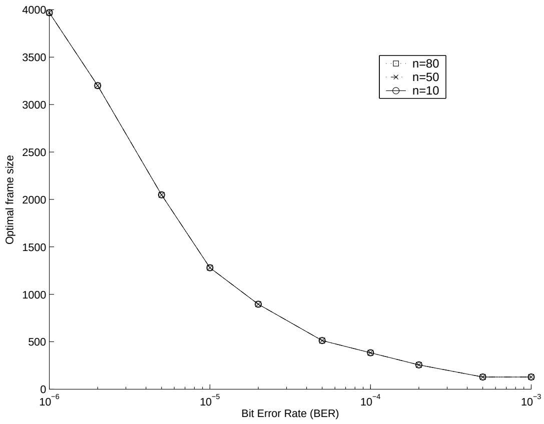

Figure 8: Optimal frame sizes at the MAC layer according to the channel conditions

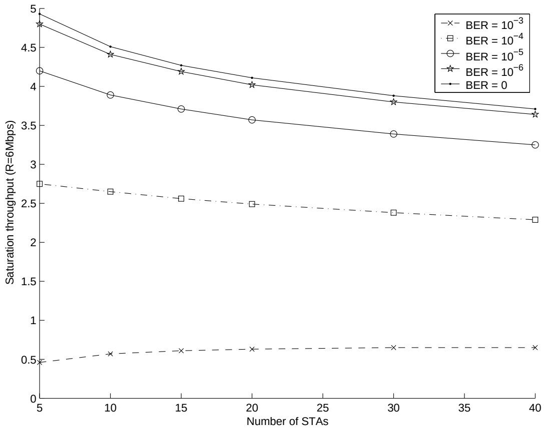

Figure 9: Saturation throughput obtained with the optimal frame size

Figure 9 shows the saturation throughput performance obtained from the model with the optimal frame sizes for different numbers of STAs. If the channel is not very noisy (i.e., when $BER \leq 10^{-4}$), 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.

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 will include 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].

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)

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)

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)