# An Analysis of the Binary Exponential Backoff Algorithm in Distributed MAC Protocols

Chunyu Hu†, Hwangnam Kim‡ and Jennifer C. Hou‡
† Department of Electrical and Computer Engineering
‡ Department of Computer Science
University of Illinois at Urbana-Champaign
E-mail:{*chunyuhu, hkim27, jhou*}*@uiuc.edu*

***Abstract*— In the paper, we perform an in-depth analytic study of the binary exponential algorithm (BEBA) that is widely used in distributed MAC protocols, for example, IEEE 802.11 DCF. We begin with a generalized framework of modeling BEBA. Then we identify a key difference between BEBA and the commonly-assumed $p$-persistent model: due to the characteristics of BEBA, the slot succeeding a busy period has a different contention rate from other slots. This causes access to a slot to be *non-uniform* and *dependent* on whether or not the slot immediately follows a busy period. We propose a detailed model with the use of a Markov chain to faithfully describe the channel activities governed by BEBA. To reduce the computational complexity, we simplify the model to an approximate one, and conduct an extensive simulation study. The analytical results derived in the proposed model are compared against those obtained from two other representative models. It is demonstrated that the proposed model is an accurate characterization of the BEBA algorithm in a broader range of system configuration.**

**We further investigate the impact of the stochastic property of the backoff time, $r$, on the performance. It is revealed that in certain circumstances it becomes an important factor that affects the performance. A case study shows that by shifting the distribution range of $r$ merely by 1 slot, substantial degradation in the system throughput may result.**

## I. Introduction

IEEE 802.11 based wireless LANs (WLANs) ascribe its conspicuous success to proliferation of portable and laptop computers and wireless-enabled PDAs, cost effective deployment of wireless networking devices, availability of the license exempt band and networking standards. In order to extend the services provided by IEEE 802.11 to beyond best efforts, various enhancement in the basic service sets have been made, including, for example, the improvement on increasing the protocol capacity, QoS extensions for delay-sensitive services, and provisioning of power report facilities. In order to assist in the incessant evolution of IEEE 802.11 protocols, an exact, analytical model that characterizes the data transmission activities governed by these protocols is important and essential.

Many research efforts have been made to devise analytic models that describe the operational properties in IEEE 802.11 wireless LANs. *Based on the devised models*, researchers then propose schemes to optimize the network capacity or to enhance the QoS in the IEEE 802.11 carrier sense multiple access mechanisms (i.e., Distributed Coordinated Function (DCF) and/or Point Coordinated Function (PCF)). An accurate, theoretically based understanding is crucial to guide the design of effective schemes. However, the nonlinear aspects inherent in the IEEE 802.11 binary exponential backoff scheme, albeit of its subtlety to impact the performance, are often ignored for analysis tractability. Moreover, as will be shown later in this paper, such an omission has practically prevented existing models from being applied to analyze IEEE 802.11 enhancements with a broader system configuration, e.g., the Enhanced Distributed Channel Access function of IEEE 802.11e (which employs a radically different contention window size as compared to DCF of IEEE 802.11).

In this paper, we present a rigorous analysis on the stationary behavior of the binary exponential backoff algorithm (BEBA) in IEEE 802.11 DCF. We begin with a generalized BEBA modeling framework. Then we identify a key difference between BEBA and the commonly-assumed $p$-persistent model: due to the characertistics of BEBA, the slot succeeding a busy period has a different contention rate from other slots. This causes access to a slot to be *non-uniform* and *dependent* on whether or not the slot immediately follows a busy period. We propose a detailed model with the use of a Markov chain to faithfully describe the channel activities governed by BEBA. To reduce the computational complexity, we simplify the model to an approximate one, and conduct an extensive simulation study. The analytical results

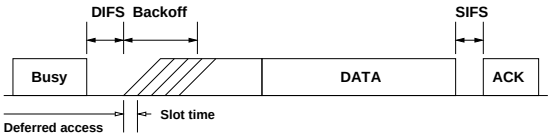



Fig. 1. IEEE 802.11 DCF without RTS-CTS.

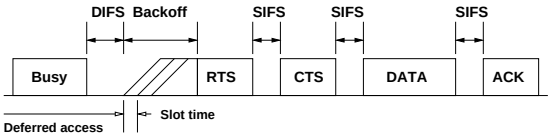



Fig. 2. IEEE 802.11 DCF with RTS-CTS.

derived in the proposed model are compared with the simulation results and those obtained from two other representative models devised by Bianchi [4] and Xiao [17]. It is demonstrated that the proposed model is a more accurate characterization of the BEBA algorithm in a broader range of system configuration.

To demonstrate the importance of considering faithfully the nonlinear aspects of BEBA, we further investigate the impact of the stochastic property of the backoff time, $r$, on the performance. We show that it is an important factor that affects the performance profoundly under certain cases. A case study shows that by shifting the distribution range of $r$ merely by 1 slot, substantial degradation in the system throughput may result.

The rest of the paper is organized as follows. In Section II, we describe the binary exponential backoff algorithm employed in IEEE 802.11 DCF and give a succinct summary of previous work on modeling BEBA. In Section III, we present a generalized theoretical framework for modeling BEBA. In Section IV, we propose a detailed model that characterizes the data transmission activities governed by BEBA. This is followed by deriving an approximate model with less computational complexity. In Section VI, we carry out a simulation study to validate and evaluate the devised models. In Section VII we investigate the impact of the stochastic property of the backoff time. Finally, we conclude the paper in Section VIII.

## II. Background and Related Work

### A. Overview of BEBA in IEEE 802.11 DCF

In distributed multiple access, a simple yet effective random backoff algorithm is widely used to avoid collisions. In particular, the binary exponential backoff algorithm [3] adjusts the contention window size dynamically in react to collision intensity. Such an algorithm is embedded in the IEEE 802.11 Distributed Coordination Function (DCF).

DCF operates as follows (Figs. 1–2). Before an attempt of data transmission is made, a station senses the channel to determine whether it is idle. If the medium is sensed idle throughout a specified time interval, called the *distributed inter-frame space* (DIFS), the station is allowed to transmit. If the medium is sensed busy, the transmission is deferred until the ongoing transmission terminates. A slotted binary exponential backoff procedure takes place at this point: a random backoff interval value is uniformly chosen in $[0, CW-1]$ and used to initialize the backoff timer, where $CW$ is the current contention window size. The backoff timer keeps running as long as the channel is sensed idle, paused when data transmission (initiated by other stations) is in progress, and resumed when the channel is sensed idle again for more than DIFS. The time immediately following an idle DIFS is slotted, with each slot equal to the time needed for any station to detect the transmission of a frame (in the IEEE 802.11 term, MAC Service Data Unit (MSDU)) from any other station. When the backoff timer expires, the station attempts to transmit a data frame at the beginning of next slot. Finally, if the data frame is successfully received, the receiver transmits an acknowledgment frame after a specified interval, called the *short inter-frame space* (SIFS), that is less than DIFS. If an acknowledgment is not received, the data frame is presumed to be lost, and a retransmission is scheduled. The value of $CW$ is set to $CW_{min}$ in the first transmission attempt, and is doubled at each retransmission up to a pre-determined value $CW_{max}$. Retransmissions for the same data frame can be made up to a pre-determined retry limit, $L$, times. Beyond that, the pending frame will be dropped.

In the case that the floor acquisition RTS-CTS mechanism is used, the same procedure is conducted except that an RTS-CTS handshake operation proceeds the DATA-ACK exchange (Fig. 2). For more details, please refer to [11].

### B. Related work

There have been a number of work on performing the saturation throughput and delay analysis of IEEE 802.11 DCF in single-hop wireless networks. They include [4], [5], [15], [6] and [12], etc. The most significant ones might be [4] done by Bianchi and [5] done by Calì *et al.*. The former uses a discrete Markov chain to model the backoff procedure performed by a tagged station, and the latter uses an iterative algorithm and takes a renewal

process view of the channel activities. They motivate substantial subsequent analysis work such as [16], [8], [17] and [13] etc.

Various modeling techniques and/or viewpoints of the system behavior are observed in these work. However, in regards to modeling BEBA, they have a lot in common, and can be nicely fitted into a framework that will be described in next section. The most common and important view (or more exactly, assumption) taken by existed work is that, the access of each station to each slot can be viewed as *independently* and *uniformly* with a common probability.

These models and the analytical results obtained have motivated and guided research efforts on capacity improvement and QoS provisioning. Among them, there are dynamical tuning the contention window size [5], model-based frame scheduling [12] and differentiation mechanisms [2].

The importance of an accurate analytical model of BEBA also lies in that it establish the theoretical bases for several extensions and/or directions such as: in-saturation throughput analysis (e.g. [9]), throughput analysis in multi-hop environments (e.g. [7]) and theoretical study of IEEE 802.11e (e.g.[17], [10] and [14]).

## III. A GENERALIZED FRAMEWORK OF MODELING BEBA

In this section we generalize a framework of modeling BEBA. A model of the backoff algorithm for distributed MAC protocols is in general composed of three steps (the first two are not necessarily conducted sequentially): First, derive the attempt probability from the backoff procedure; Second, model the transition of the channel state; and lastly, interpret the stationary channel state probabilities obtained into results wanted according to specific protocol details. We will takethroughput as an example. It is also possible to derive other quantitities such as delay and packet dropping ratio (e.g. [17]).

### A. Assumptions

The network we are interested is a single-hop wireless network, composed of $N$ stations. They share one channel and the access is governed by the afore-described binary exponential backoff procedure. All stations can hear each other, which implies that there is no hidden terminal. It is possible to extend the model and take the hidden terminal problem into consideration, however, as done in [5]. Besides, the RTS-CTS virtual sensing mechanism can help alleviate the hidden terminal issue. The channel is an ideal one, introducing no errors to the reception of a packet other than collisions. The capture technique is not considered.

The network is in the saturation condition. That is, every station is backlogged and always has a packet to deliver.

The backoff algorithm is performed in a time-slotted fashion as described in Section II-A and the backoff timer is counted in integer number of slots. Note that a station attempts to access only at the beginning of a slot, and during the time when there are channel activities (a successful transmission or a collision), all stations not participating the activity will freeze their backoff timers (enabled by the physical/virtual sensing mechanism). Therefore, a busy period, which refers either a successful transmission or a collision, can be deemed as a virtual time slot. Now we can practically view the channel in discrete time slots and all stations are well-synchronized in time slots. For the latter to be reasonable, it is necessary that compared to the length of an idle slot the propagation delay is negligible. Indeed, IEEE 802.11 regulates the slot length to be $20\mu s$ and the maximum propagation delay in a system with $300m$ communication range is $1\mu s$.

We say the channel state in a slot is *idle* (I) (resp. *success* (S), *collision* (C)), if by end of the slot, no (resp. one, more-than-one) station(s) have transmitted during that slot. In addition, a *busy* (B) slot refers to either a success or a collision slot.

### B. Attempt probability

Deriving the attempt probability and modeling the channel state transition are two closely related steps. There is an important assumption made in the literature that, each station attempts to access each slot *independently* and *uniformly* with a probability, which we term as the *attempt probability*, denoted by $p$. Essentially, this maps the backoff algorithm into the $p$-persistent scheme. All existed work to our awareness share this view and differ mainly in the approach of deriving $p$ from the backoff procedure.

One of the first explorers, Bianchi [4] uses a Markov chain to characterize the backoff procedure, represented by the *backoff stage* and the *backoff counter value*. Solve the equilibrium equation for the Markov chain, and one can express the attempt probability $p$ in terms of a collision probability that can be derived from the next step. Using an iterative equation yields $p$.

Two major extensions are made to Bianchi's model. One (e.g. [17]) is to incorporate the retry limit $L$ into the model as the highest backoff stage. In the other one (e.g. [8] and seen in [17]) a self-loop is added to each

state in the Markov chain to reflect that the backoff counter is frozen during the time when there are channel activities.

There are several work that stem from Bianchi's model but try to avoid analyzing the Markov chain to obtain $p$ (e.g. [13]); or that try to simplify the expression of $p$ in terms of the contention window size (e.g. [6]).

Another representative work is done by Calì, *et al.* [5] (referred as Calì's model). A relationship of $p$ and the current contention window size $CW$ is established first. Then $p$ is derived using an iterative algorithm that considers the dynamics of $CW$ and computes the average contention window size $E[CW]$. We note that in case that $CW_{min} = CW_{max}$ and the contention window size is uniformly $CW$, both Bianchi's model and Calì's model yield $p = \frac{2}{CW+1}$.

### C. The stationary channel state probabilities

With the assumption of the $p$-persistent access pattern, the distribution of the channel states can be derived independently for each slot. Denote the stationary probability that a slot is idle, success and collision by $P_I$, $P_S$ and $P_C$, respectively. If $p$ is known, one can obtain the three probabilities by:

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

$$P_S = Np\,(1-p)^{N-1} \tag{2}$$

$$P_C = 1 - (1-p)^N - Np\,(1-p)^{N-1} \tag{3}$$

### D. Throughput

Knowing $[P_I\ P_S\ P_C]$, we next consider the actual length of a success and collision slot in order to derive the throughput. Denote them by $T_D$ and $T_C$, respectively. In the basic access of IEEE 802.11 DCF, a successful transmission includes the transmission of the DATA frame, a SIFS and an ACK transmitted by the receiver. Because after each successful transmission, the backoff timer is resumed after a DIFS idle period, we counter DIFS in each successful transmission to ease the computation of idle time. Therefore $T_D = DATA+SIFS+ACK+DIFS$. A collision is awared by the transmitter upon a sender timer timing-out, and is detected at other stations by receiving corrupted packets. After detecting a collision, the receiver node resumes the backoff timer after an EIFS idle period to avoid preempting the channel over transmitter stations. $EIFS = SIFS + ACK + DIFS$. Therefore, the colliding and non-colliding stations resume their backoff timers at approximately the same time. This gives $T_C$ as $T_C = DATA_{max} + SIFS + ACK + DIFS$, where $DATA_{max}$ is the longest DATA frame in the collision. In case that the RTS-CTS handshake is used, $T_D$ and $T_C$ can be similarly developed. Denote the payload in a DATA packet by $payload$ and the length of an idle slot by $aSlotTime$. The throughput of the network is obtained by:

$$\eta = \frac{P_S \cdot payload}{P_I \cdot aSlotTime + P_S T_D + P_C T_C} \tag{4}$$

A different view of the channel activities is taken by Calì *et. al* in [5]. For a successful transmission to occur, the channel typically experiences several alternating occurrences of an idle period and a collision period and lastly an idle period. This is defined as a virtual transmission time, $t_v$, and is viewed as a renewal process. Formally,

$$t_v = E[N_c] \cdot T_C + E[idle]\ (E[N_c]+1) + T_D \tag{5}$$

where $N_c$ is the number of collisions before a successful transmission occurs and $idle$ is the length of consecutive idle slots. Each idle period can be viewed as a renewal process, too. Denote the number of consecutive idle slots plus 1 by $b$. Both $N_c$ and $b$ are subject to the Negative Binomial distribution, with parameter $(1, \frac{P_C}{P_S})$ and $(1, 1-P_I)$, respectively. Hence,

$$E[N_c] = \frac{P_S}{P_C} \ \text{and}\ \ E[idle] = \frac{P_I}{1-P_I}\, aSlotTime \tag{6}$$

Bring (6) back to (5) and we obtain $t_v$. The throughput can be computed by averaging the payload over $t_v$. The result shows that it has the same mathematical form as the throughput derived by Bianchi's model (Eqn. (4)).

By now, we have generalized a framework of modeling BEBA, which can accommodate the work previously proposed. The essential, common assumption made by the literature is that the channel access governed by BEBA can be modeled as $p$-persistent. It has been shown that two representative approaches ([4] and [5]) have equivalent mathematical form given a uniform contention window size (but they do provide different and insightful views of the problem).

## IV. Modeling the Binary Exponential Backoff Algorithm

Recall that in the backoff procedure, at the beginning of each slot, a station transmits if its backoff timer has expired by that time. Otherwise, depending on the channel state (idle or busy), by end of the current slot, the station will count down the backoff counter by 1 or will have frozen it at a value. Suppose the current slot ends in the busy state. When next slot begins, the stations naturally fall into one of the two groups: those that

have transmitted, and those who did not and have frozen their backoff counters. Ideally the backoff counter does not freeze at 0. This introduces a short-term unfairness among the two groups of stations: only stations in the former group have privileged possibility to access this slot since after transmitting these stations generate a new backoff time with a chance to be 0. For convenience, we refer a slot subsequent to a busy slot as a *post-busy* slot.

This subtle yet crucial different access behavior in the post-busy slot leads the backoff algorithm deviate the $p$-persistent model commonly assumed in previous work. Our model will capture this feature.

### A. The channel state transition

We make the same assumptions as given in Section III-A. In brief, a network composed of $N$ stations that share one channel in ideal condition and they are operated in the saturation condition. There are three channel states in a slot: *idle*, *success* and *collision*. In this section, we differentiate collision states by the number of stations that transmit and incur the collision. More generally, we define the channel state space as $\mathbb{S} = \{B_k, k = 0, 1, \cdots, N\}$, where $B_k$ refers the channel state in a slot such that $k$ stations transmit in that slot. Note that $I \equiv B_0$, $S \equiv B_1$ and $C \equiv \bigcup_{k\geq 2}\{B_k\}$.

We model the system using a discrete Markov chain (Fig. 3 (a)) in the expanded channel state space $\{B_k, k = 0, 1, \cdots, N\}$. The channel state is sampled at the end of each slot. The access to a post-busy slot and that to slots thereafter (in another word, a slot subsequent to an idle slot) are made with different probabilities. For a post-busy slot, first, only those who have transmitted during the preceding busy slot will possibly access again; second, a station among them access the slot with probability $\frac{1}{CW}$. The later is because an eligible station can access the post-busy slot only if it chooses 0 among $[0, CW - 1]$ as the new backoff time. Since the backoff time is uniformly generated, the probability of choosing 0 is $\frac{1}{CW}$. $CW$ is the current contention window size. We will soon develop the derivation of the average contention window size $E[CW]$ and use it in place of $CW$. For slots other than post-busy slots, we assume it is accessed uniformly and independently with probability $\tau$.

Note that in the Markov chain, a direct transition is possible only if it is directed from a state to another with the latter having an equal or smaller subscript (except that a transition can originate from state $B_0$ to any other states). It is because the post-busy slot is contended by only those have transmitted in the previous busy slot. The non-zero one-step transition probabilities are given by

$$\begin{cases} p_{0,k} &= \binom{N}{k}\tau^k(1-\tau)^{N-k}, \\ p_{k,j} &= \binom{k}{j}\left(\frac{1}{CW}\right)^j\left(1-\frac{1}{CW}\right)^{k-j}, \end{cases} \quad \text{where k} = 0\cdots N \text{ and j} = 0\cdots k \tag{7}$$

Denote the transition probability matrix by $\mathbb{P}$. Assume that $\tau$ is known (we will derive it in Section V), then all the elements in $\mathbb{P}$ can be calculated. The equilibrium state of the Markov chain $\pi = [\pi_0\ \pi_1\ \cdots\ \pi_n\ ]$ can be solved from the equilibrium equation $\pi = \pi\mathbb{P}$ using Matlab. Recall the stationary probabilities that a slot is idle, success and collision are denoted by $P_I$, $P_S$ and $P_C$, respectively. We have obtained $P_I = \pi_0$, $P_S = \pi_1$ and $P_C = \sum_{k\geq 2}\pi_k$.

Following the third step as described in the generalized framework, the system throughput can be computed accordingly using (4).

### B. A simplified and approximate model

The state space of the model proposed above has is of $(N + 1)$-dimension and is not scalable in terms of $N$. This motivates us to simplify the model to fit into a more tractable state space. As shown by Fig. 3 (b), the states $\{B_k, k \geq 2\}$ are merged into a single state, $C$ (a.k.a. the collision state).

The transition probabilities from the idle state and the success state to other possible states are respectively

$$\begin{cases} p_{ii} &= (1-\tau)^N \\ p_{is} &= N\tau(1-\tau)^{N-1} \\ p_{ic} &= 1 - p_{ii} - p_{is} \end{cases} \tag{8}$$

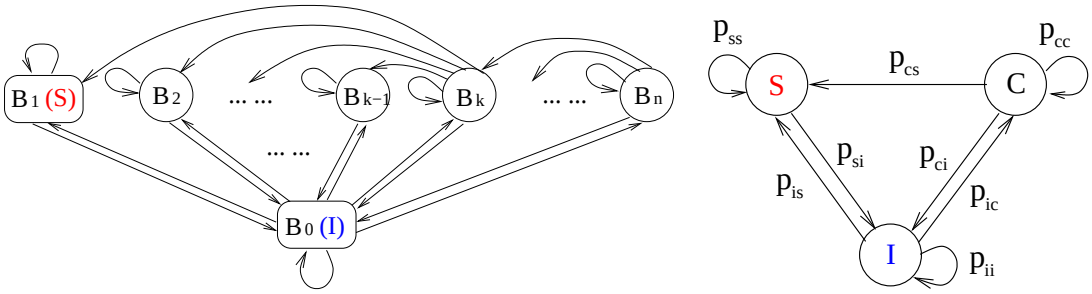
and

$$\begin{cases} p_{ss} = \frac{1}{CW} \\ p_{si} = 1 - \frac{1}{CW} \end{cases} \tag{9}$$

To derive the transition probability from the collision state to the idle state, we condition on the event that $m$ $(m = 2\cdots N)$ stations transmit in the collision slot. Denote this event by $A_m$.

$$\begin{aligned} p_{ci} &\triangleq \text{P [an idle slot follows a collision]} \\ &= \sum_{m=2}^{N} \text{P [an idle slot follows a collision } |A_m] \\ &\quad \times \text{P } [A_m|\text{a slot is a collision slot}] \end{aligned}$$

Given $A_m$, the probability that an idle slot follows a collision is simply the probability that none of these $m$ stations choose 0 as the new backoff time, that is, $(1 - \frac{1}{CW})^m$. However, the conditional probability of the event




(a) The detailed Markov chain (b) The simplified Markov chain

Fig. 3. Discrete Markov chains for the channel state sampled at the end of a slot. The subscript $k$ denotes the number of stations that transmit in the slot. I – Idle, S – Success, C – collision.

$\{A_m|$ a slot is a collision$\}$ is difficult to obtain because of the transitions among detailed collision states $\{B_k, k \geq 2\}$ as shown in Figure 3(a). We use the following to approximate this probability.

$$\begin{aligned} &\text{P}\ [A_m|\ \text{a slot is a collision slot}] \\ &= \binom{N}{m}\tau^m(1-\tau)^{N-m}\frac{1}{p_{ic}} \end{aligned} \tag{10}$$

With some manipulation of the equation, one can derive $p_{ci}$ as:

$$p_{ci} = \frac{1}{p_{ic}}\left[\left(1-\frac{\tau}{CW}\right)^N - p_{ii} - \left(1-\frac{1}{CW}\right)p_{is}\right] \tag{11}$$

Make a similar approximation and one can obtain:

$$\begin{aligned} p_{cs} =& \sum_{m=2}^{N} m\frac{1}{CW}\left(1-\frac{1}{CW}\right)^{m-1} \\ &\times \binom{N}{m}\tau^m(1-\tau)^{N-m}\frac{1}{p_{ic}} \\ =& \frac{N\tau}{CW}\cdot\frac{1}{p_{ic}}\cdot\left[\left(1-\frac{\tau}{CW}\right)^{N-1} - p_{ii}\right] \end{aligned} \tag{12}$$

and $p_{cc} = 1 - p_{ci} - p_{cs}$.

The stationary probabilities $[P_I\ P_S\ P_C]$ can be obtained by solving the equilibrium equation for this simplified Markov chain.

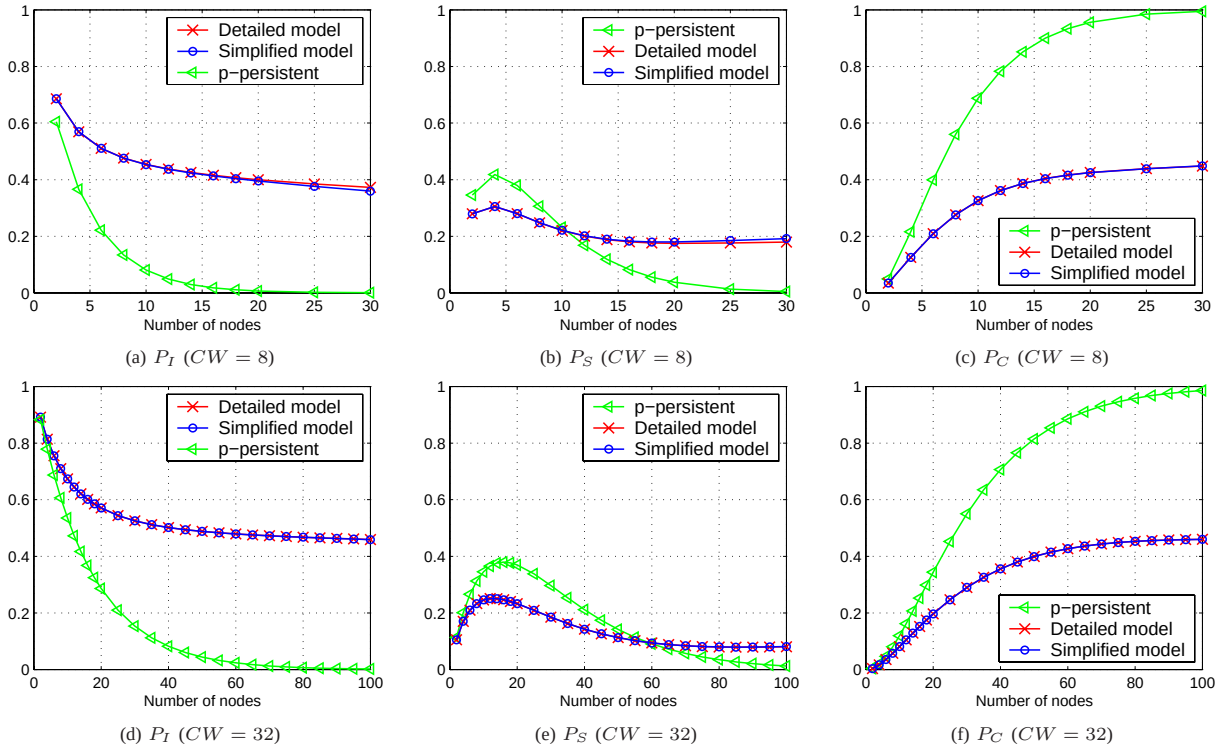
**Remark:** Note that $\lim_{CW\to\infty}\frac{1}{CW} = 0$. In the simplified model, as $CW \to \infty$, $\{p_{ci},\ p_{si}\} \to 1$ and $\{p_{cc},\ p_{cs},\ p_{ss}\} \to 0$; and the model falls back to the $p$-persistent model as assumed in all previous work – all stations contend the access to each slot *independently*, and *uniformly* with a probability. Therefore previously models can be viewed as a special case of the proposed model.

### C. Numerical results of $[P_I\ P_S\ P_C]$

We term the models described in Section IV-A and Section IV-B as the *detailed model* and the *simplified model*, respectively. To compare the proposed models and previous models (generally referred as the $p$-persistent model), and show how well the simplified model approximates the detailed one, we compute the numerical results of $[P_I\ P_S\ P_C]$ obtained from each of the three models. A uniform contention window size $CW$ is assumed. In our model, the attempt probability is given by $\tau = \frac{2}{CW}$ as will be explained in Section V. In the $p$-persistent model, the attempt probability is given by $p = \frac{2}{CW+1}$. Figure 4 presents two sets of numerical results with $CW = 8$ and 32, respectively.

Three key observations are made. First, the simplified model is a good approximation of the detailed model. The difference between the results generated by them is slight and is noticeable only when the number of nodes increases beyond 20 in the case $CW = 8$ as shown in Fig. 4(a) and (b). Even though, the difference of $P_I$'s at $n = 30$ and $CW = 8$ is merely around $0.012$. Since $[P_I\ P_S\ P_C]$ determines the throughput and the results of the simplified model closely match those of the detailed model but is computationally simpler, we will assume the simplified model in later sections.

Second, the differences in the results of the proposed models and the $p$-persistent model are significant. The differences in $P_I$'s and $P_C$'s obtained from the two models increase as the number of stations increases. The difference in $P_S$'s shows a non-monotone trend. It first increases as $N$ increases, then shrinks till zero (e.g. at $n = 55$ in Fig. 4(e)) and then keeps decreasing (in the inversed direction). These pronounced discrepancies indicate that the proposed models predict a significantly different pattern of the channel activities, such as longer idle time and less collisions. One of the implications is that the energy consumption computed from the two




(a) $P_I$ $(CW = 8)$ (b) $P_S$ $(CW = 8)$ (c) $P_C$ $(CW = 8)$

(d) $P_I$ $(CW = 32)$ (e) $P_S$ $(CW = 32)$ (f) $P_C$ $(CW = 32)$

Fig. 4. Compare the numerical results of $[P_I P_S P_C]$ obtained from three models: the detailed model, the simplified model and the $p$-persistent model. A uniform contention window size $CW$ is assumed.

types of models might be rather different.

Third, the maximum $P_S$ as predicted by the $p$-persistent model is higher than that of the proposed model. The former is around $0.4$, unsurprisingly close to the theoretical capacity of slotted-ALOHA ($0.368$). The latter, however, is only around $0.3$ at $CW = 8$ and $0.25$ at $CW = 32$. After exceeding the maximum $P_S$, both curves begin to drop. However, the $P_S$ of the proposed model falls at a much slower rate than that of the $p$-persistent model; it finally exceeds the latter and maintains this superiority afterwards. The reason behind this trend is that when $N$ is relative small, the post-busy slot is likely to be idle and lowers the throughput. However, when $N$ grows, the overall contention intensity is high, leaving little chance for a successful transmission. But the post-busy slot is contended by only a subset of all nodes and thus has larger chance for a successful transmission. Indeed, as we observe in the simulation, when $N$ is very large relative to the contention window size, most successful transmissions take place in the post-busy slot. This reveals a fundamental difference between the backoff algorithm and the $p$-persistent scheme.

## V. Derivation of the attempt probability $\tau$

### A. Relation between $\tau$ and $CW$

Let us start with the simple case that all stations have a uniform contention window size $CW$. Recall that $\tau$ is the probability that a station transmit at the beginning of a slot subsequent to an idle slot. It measures the possibility that a station's backoff counter reaches zero at the beginning of a non- post-busy slot. Denote a station's backoff counter at the beginning of such a slot by $r_1$ ($r$ is saved to denote a backoff time newly generated). It is reasonable to assume $\tau = \frac{1}{E[r_1]+1}$, where 1 in the denominator counts the slot in which the station transmits. Because $r$ is uniformly polled from $[0, CW-1]$, $r_1$ can be regarded as being approximately uniformly distributed in $[0, CW-2]$. Hence the relation between $\tau$ and $CW$ is established as

$$\tau = \frac{1}{E[r_1]+1} = \frac{2}{CW} \tag{13}$$

### B. An iterative algorithm to derive $\tau$

More generally, to alleviate collision intensity, the binary exponential backoff algorithm requires the contention window size varies in $[CW_{min}, CW_{max}]$ gov-

erned by the procedure as described in Section II-A. An iterative algorithm is developed to derive the average contention window size, $E[CW]$.

We take the view of a tagged station. Denote the contention window size in the $i-$th iteration is $CW^{(i)}$ and correspondingly, $\tau^{(i)} = \frac{2}{E[CW^{(i+1)}]}$. When the station of interest transmits, the probability that it experiences a collision is

$$p_c^{(i)} = 1 - \left(1 - \tau^{(i)}\right)^{(N-1)} \tag{14}$$

With the collision probability $p_c^{(i)}$, the mass probability of $CW^{(i+1)}$ is given by, for $k = 0, \cdots, L$,

$$q_k \stackrel{\triangle}{=} P[CW^{(i+1)} = W_k] = p_0 \left(p_c^{(i)}\right)^k \tag{15}$$

where $p_0$ is a normalization factor and can be computed by noting $\sum_{k=0}^{L} q_k = 1$. $L$ is the retry limit and $W_k = \min\{2^k CW_{min}, CW_{max}\}$.

This iterative algorithm extends the algorithm proposed in Calì's model and incorporates the retry limit. The convergence of the iterative algorithm can be proved similarly to those in [5] and [12].

### C. Attempt probability and transmission probability

We use the *attempt probability* in both our model and previous models to refer the probability that a station transmits at the beginning of a slot (of course, in our model, this slot is limited by excluding the post-busy slot). We use a general notation here, $p_a$. More precisely, this probability measures how likely a station's backoff timer expires at that moment. Hence $p_a$ depends on the value that the backoff timer holds at the beginning of each slot. This value, when being generated, is uniformly polled from the contention window, and is decremented by 1 for each *idle* slot. For each elapsed *busy* slot, note that, the counter is simply frozen, *instead of being prolonged with the probability that the slot is sensed to be busy.* Therefore, $p_a$ solely depends on the contention window size $CW$. In our model,

$$p_a \equiv \tau = \frac{1}{E[r_1] + 1} \tag{16}$$

and in the previous models,

$$p_a \equiv p = \frac{1}{E[r] + 1} \tag{17}$$

where $E[r_1] = E[r] - \frac{1}{2}$.

The *transmission probability* in this section particularly refers to the term frequently used in a general $p$-persistent model. It is the probability that a station transmits in *any* slot. A way to measure it is to count the number of transmission during a certain period and average the result over the total number of slots in that period of time.

Put in the context of the slotted backoff procedure, to measure or estimate the transmission probability, we count the number of slots that elapse before a station is allowed to transmit. Term this amount of time by *waiting time* $T$ (in slots). Required by the backoff algorithm, a station has to wait $r$ *idle* slots (recall $r$ is the number of backoff slots that a station generates) before it can transmit. Consider that it is possible that the channel becomes busy during the waiting time $T$ (with probability $P_B = 1 - P_I$), we conclude that $T$ is a Negative Binomial random variable with parameters $(r, p_b)$. Its mean is given by:

$$E[T] = E[\, E[T|r]\,] = \frac{E[r]}{p_b} \tag{18}$$

The transmission probability is estimated by:

$$p_{tx} = \frac{1}{E[T] + 1} = \frac{1}{\frac{E[r]}{p_b} + 1} \tag{19}$$

Now it is clear that the transmission probability $p_{tx}$ relies on both the contention window size and the number of stations.

Examining the models proposed for BEBA, we note that $p_a$ is used in most work (in particular, Bianchi's and Calì models) though appears in various terms such as *the probability that a station transmits in a randomly chosen slot time* in [4], *transmission attempt* in [5] and *attempt rate* in [12] and [13].

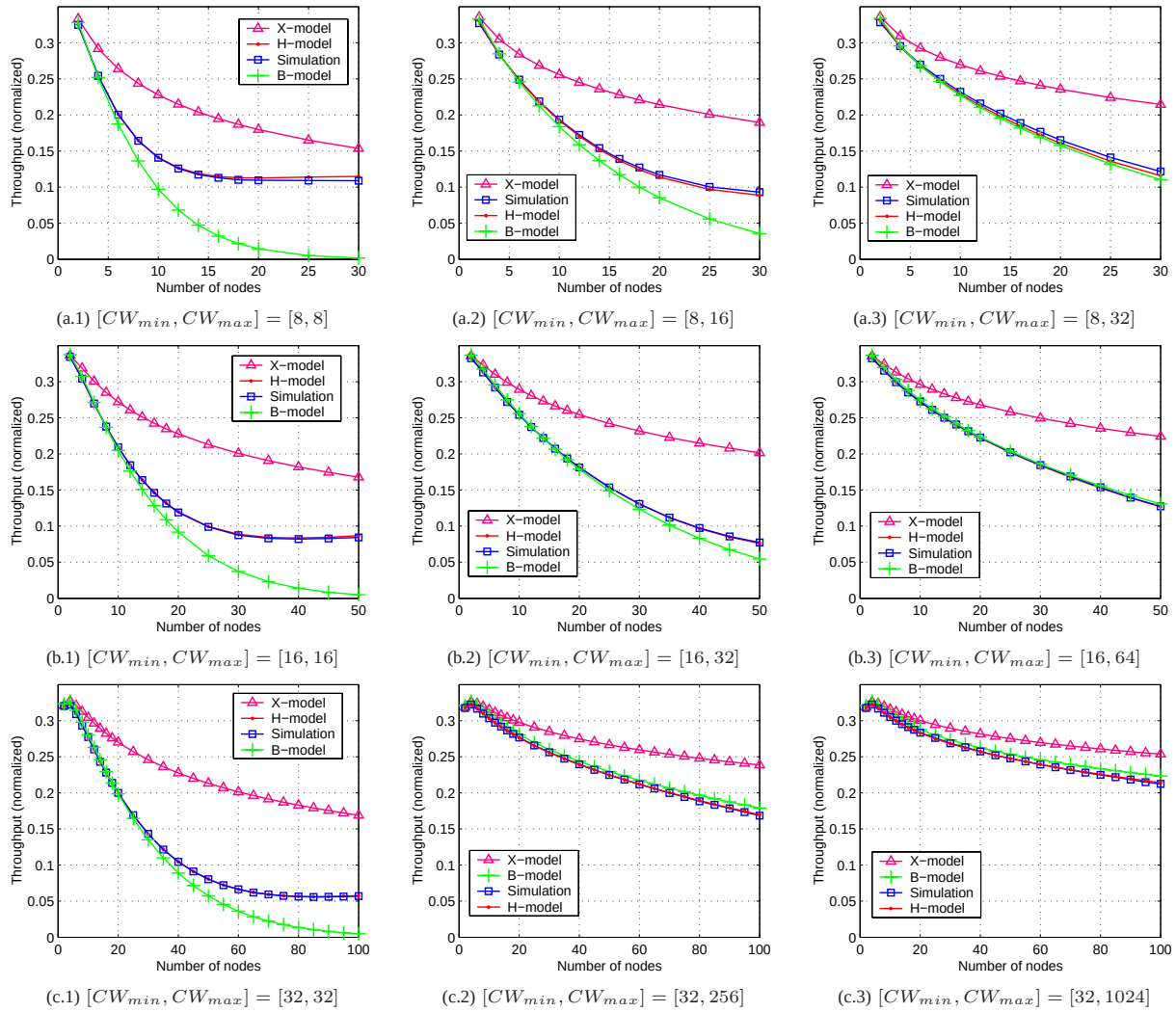
The only exceptions are seen in an extension made to Bianchi's model (e.g. [8] and [17]), in which $p_{tx}$ is used instead. In this extension, a self loop is added to each state in the Markov chain to reflect that the backoff timer is frozen during the time that the channel is busy. Derived from Equations (5-7) in [17], *"the probability that a station transmits during a generic slot time"* is

$$\tau = \frac{1}{1 + \frac{CW-1}{2(1-p)}} \tag{20}$$

where $p$ is the probability that the channel is idle and thus $1 - p = p_b$. By noting that $E[r] = \frac{CW-1}{2}$, (20) has exactly the same form as (19).

## VI. Model Validation

We have performed extensive simulation over a large range of parameters to validate the proposed model and to compare it with two other models: Bianchi's model as presented in [4] and an extension made to Bianchi's




(a.1) $[CW_{min}, CW_{max}] = [8, 8]$ (a.2) $[CW_{min}, CW_{max}] = [8, 16]$ (a.3) $[CW_{min}, CW_{max}] = [8, 32]$

(b.1) $[CW_{min}, CW_{max}] = [16, 16]$ (b.2) $[CW_{min}, CW_{max}] = [16, 32]$ (b.3) $[CW_{min}, CW_{max}] = [16, 64]$

(c.1) $[CW_{min}, CW_{max}] = [32, 32]$ (c.2) $[CW_{min}, CW_{max}] = [32, 256]$ (c.3) $[CW_{min}, CW_{max}] = [32, 1024]$

Fig. 5. Comparing the saturation throughput of analytical results and simulation results. DATA payload = 500 bytes.

| | |
|---|---|
| $t_{slot}$ | 20 $\mu s$ |
| SIFS | 10 $\mu s$ |
| DIFS | 50 $\mu s$ |
| Data Rate | 11 Mbps |
| PLCPDataRate | 1 Mbps |
| PreambleLength | 144 bits |
| PLCPHeader | 48 bits |
| MacHeader | 28 bytes |
| ACK payload | 14 bytes |

TABLE I

SYSTEM CONFIGURATION USED IN THE SIMULATIONS

model in [17][1]. We use the initial of the first author as the prefix and they are abbreviated as *B-model* and *X-model*, respectively. The simplified model we have proposed is assumed and referred as *H-model*.

We use a simulator written in C to emulate the IEEE 802.11 MAC DCF in the saturation condition. The code is available at [1] [2]. The system configuration used in the simulations are listed in Table I. The retry limit is set to 7. The DATA frames are of uniform size. We have simulated with the DATA packet payload varying from 100 - 1000 bytes and the trends observed are similar. In the results presented, the payload is 500 bytes.

[1]The paper studies the saturation throughput and delay for differentiated services. We let the number of classes be 1.

[2]The simulator is for pure purpose of analyzing the 802.11 MAC DCF performance under the saturation situation. It emulates the idealized slotted access governed by the binary exponential backoff algorithm. Its validity has been confirmed by comparing to *ns-2* results.

We vary the contention window size range and plot the throughputs (normalized over the bandwidth) obtained from both theoretical analysis and simulation results in Figure 5. Group (a), (b) and (c) in the figure correspond to the cases: $\{CW_{min} = 8, h = 0, 1, 2\}$, $\{CW_{min} = 16, h = 0, 1, 2\}$, and $\{CW_{min} = 32, h = 0, 3, 5\}$, respectively. $h = log_2 \frac{CW_{MAX}}{CW_{MIN}}$. The three groups of figures demonstrate common trends.

First, the results obtained from the model we have proposed match the simulation results accurately with the confidence intervals of 95%, 97% and 99% for $CW_{min} = 8$, 16 and 32, respectively.

Second, the results obtained from Binachi's model fit the simulation results well when the number of stations is small relative to the (average) contention window size, CW. This is in particular obvious in Fig. 5 (b.3), (c.2) and (c.3) when $CW$ falls in a higher range. The reason has been analyzed in Section IV-B. In other figures, the initial (left) parts of the two curves are close to each other. As $N$ increases, however, the curves of Bianchi's model gradually depart from those of simulation results, and show a pessimistic trend.

Third, the extension of Bianchi's model (referred as X-model in Fig. 5) is in general too optimistic in estimating the throughput. The results obtained by it are significantly higher than the simulation results. The differences generally increase as there are more stations and as large as twice of the simulation results, for example, at $N = 20$ in Fig. 5 (b.1). The reason is that the transmission probability as the prolonged attempt probability is used in computing the stationary channel state probabilities.

**Remark:** The simulation results show that our model captures the characteristics of the backoff algorithm faithfully and is applicable to a broad range of protocol and system parameters (small $CW$ and large $N$). In contrast, as done in all previous work, assuming uniform access of each slot fails to capture all of the important features of the backoff algorithm.

## VII. Impact of the Stochastic Property of the Backoff Time

When a station's backoff counter is decremented to zero, a new backoff time ($r$ slots) will be generated. IEEE 802.11 DCF requires $r$ be uniformly distributed in $[0, CW - 1]$. The model proposed gives us a hint that the stochastic property of the backoff time may affect the performance. For a simple example, if the chance of choosing $r = 0$ is large than other values while the average holds the same, intuitively the contention in the post-busy slot will increase substantially.

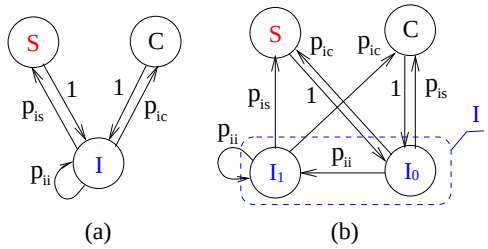



Fig. 6. Discrete Markov chains for the channel state sampled at the end of a slot. They model the design that the backoff slots $r$ is uniformly distributed in $[1, CW]$. (a) and (b) are equivalent, and (b) provides more details on the idle state. $I_0$ – Idle in a post-busy slot, $I_1$ – Idle in a slot subsequent to an idle slot.

We are interested in how much is the impact of the stochastic property of $r$. We have a case ready as our subject. In the literature, a slightly different $r$ is often mentioned (e.g. two recent work, [14] and [13]). That is, $r$ is uniformly distributed in $[1, CW]$ instead of $[0, CW - 1]$. Obviously, such a shift in the distribution range increases the mean by 1, which alters the attempt probability a little; and the understanding on its implication is limited as this far to our awareness.

Guided by the model proposed for the design $r \in [0, CW - 1]$ uniformly, we build a model for this seemingly very alike random variable $r \in [1, CW]$ uniformly and investigate the performance.

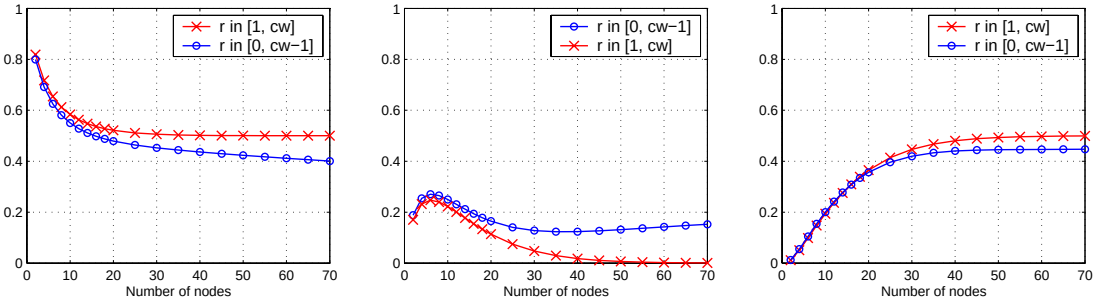
### A. A case study: Model BEBA with $r \in [1, CW]$ uniformly

With the backoff time $r$ uniformly distributed in $[1, CW]$, a backoff time newly generated is always no less than 1. Consequently, in a post-busy slot, no stations will attempt to transmit – stations that have not transmitted in the busy slot freeze their backoff counters with remaining time at least 1 slot; and station(s) that transmit in the busy slot will not, either. Therefore the channel state in a post-busy slot is idle with probability 1. We designate this channel state as $I_0$. After this post-busy slot, stations resume the contention for the channel access. As before, we assume they contend with a uniform probability $\tau$. With the shift of the distribution range of $r$, the computation of $\tau$ from $CW$ is changed to:

$$\tau = \frac{2}{CW + 1} \tag{21}$$

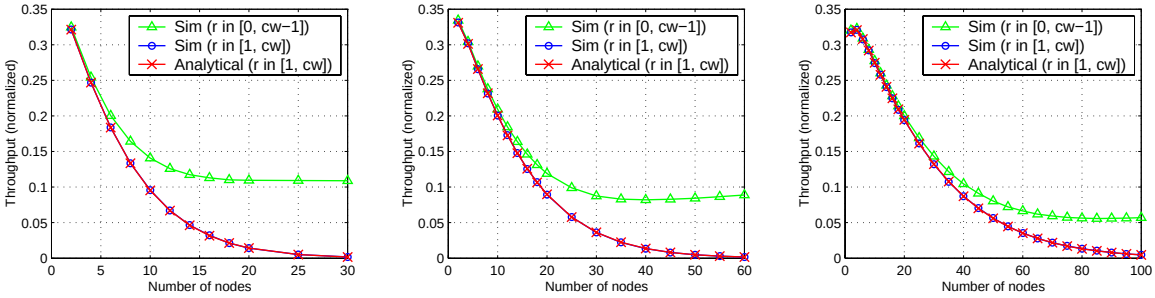
In case $CW_{min} \neq CW_{max}$, we can derive $E[CW]$ using the iterative algorithm described in Section V-B.

The above behavior can be modeled by a Markov chain, as shown by Fig. 6 (a). State $I$, $C$ and $S$ bear




(a) $P_I$ ($CW = 16$) (b) $P_S$ ($CW = 16$) (c) $P_C$ ($CW = 16$)

Fig. 7. Compare the numerical results of $[P_I P_S P_C]$ obtained from two designs: $r$ in $[0, CW-1]$ and $r$ in $[1, CW]$. A uniform contention window size $CW = 16$ is assumed.




(a) $CW = 8$ (b) $CW = 16$ (c) $CW = 32$

Fig. 8. Comparing the saturation throughput achieved by two designs: $r \in [0, CW-1]$ and $r \in [1, CW]$. DATA payload = 500 bytes.

the same meanings as in Fig. 3 (b). The equilibrium equation for the Markov chain can be manually solved and the stationary probabilities are:

$$[P_I \ P_S \ P_C] = \left[\frac{1}{2-p_{ii}} \ \frac{p_{is}}{2-p_{ii}} \ \frac{p_{ic}}{2-p_{ii}}\right] \qquad (22)$$

Note that $P_I$ is lower-bounded by $0.5$ and the lower-bound is obtained at $p_{ii} = 0$, which corresponds to $N \to \infty$. It further interests us the expenditure of the time in idle states. In Fig. 6 (b) idle states are categorized into two types: $I_0$ and $I_1$. $I_0$ has been defined in the above. State $I_1$ refers to the idle state in a slot subsequent to an idle slot. From the Markov chain, we can obtain the stationary probabilities of states $I_0$ and $I_1$:

$$P_{I_0} = \frac{1}{2-p_{ii}} \quad \text{and} \quad P_{I_1} = \frac{p_{ii}}{2-p_{ii}} \qquad (23)$$

As $N \to \infty$, $p_{ii} \to 0$, and it leads to $P_{I_0} \to 0.5$ and $P_{I_1} \to 0$. Clearly, $P_{I_0}$ attributes to the constant part of the lower-bound of $P_I$. This means even $N$ is very large, half of the time (in the sense of equal virtual time slot length) is totally wasted – during which no stations even attempt to transmit. It is one of the pitfalls that we shall avoid in designing and configuring a network system.

Fig. 7 depicts the numerical results of $[P_I \ P_S \ P_C]$ at $CW = 16$ and compares them with those obtained from the design that $r$ is uniformly distributed in $[0, CW-1]$. Fig. 7 (a) visualizes the trend that as $N$ increases, the probability that a slot is idle approaches $0.5$.

Another trend to our concern is the change of the success probability $P_S$. If all the successful transmissions and collisions have the same actual duration as an idle slot, $P_S$ is approximately (due to the PHY/MAC headers) the actual throughput. As shown in Fig. 7 (b), $P_S$ drops in a much faster rate than in the case that $r$ is uniformly distributed in $[0, CW-1]$. All the observations indicate that the performance is significantly impaired by shifting the range of $r$ by 1. Next, we will evaluate the actual throughput with data payload of 500 bytes (thus the transmission/collision period is approximately 29 slots) by means of both theoretical analysis and simulation.

### *B. Performance evaluation*

To evaluate the impact of the change in $r$ on the performance, we compare the throughput obtained from simulations for both designs. Meanwhile, to validate the model described in the above, the analytical results are

plotted, too.

Three sets of results are shown in Fig. 8 (a-c) with uniform contention window size $CW = 8$, 16, and 32, respectively. Several observations are made.

First, the curves of the analytical results are almost coincident on those of the simulation results. This shows that the model accurately captures the system characteristics in this design.

Second, the throughput obtained by the design $r \in [1, CW]$ is significantly lower than that of the design $r \in [0, CW - 1]$, especially when $N$ is large relative to $CW$. This performance degradation becomes more evident as $N$ increases. For example, as shown in Fig. 8 (a), when $N$ reaches 20, the throughput achieved by the design $r \in [0, CW - 1]$ drops to $1.2$. In contrast, the design $r \in [1, CW]$ achieves the throughput only nearly $0.16$, approximately $12.5\%$ of the former.

In summary, we have witnessed that shifting the distribution range of $r$ merely by 1 results in a great performance degradation.

## VIII. Conclusion

In this paper, we have devised an analytic model that characterizes the data transmission activities governed by the binary exponential backoff algorithm (BEBA). We observe that access to a slot becomes non-uniform under BEBA. Nodes that have transmitted in a slot have a second chance to access the subsequent slot, while the other stations do not have this privilege. Consequently, access to a slot is not totally independent among slots. We devise a discrete-time Markov chain to characterize the transition of the channel state, taking into account of the above subtlety. The numerical results show that the BEBA exhibits different properties from the $p$-persistent model (which is commonly used to characterize the BEBA). For example, the maximum stationary success probability $P_S$ achieved under the BEBA is lower than that under the $p$-persistent model. The simulation study has validated the proposed model and confirmed the above findings. With the proposed models, we study the impact of the stochastic property of the backoff time $r$ on the performance. Both our analytical and simulation results show that the subtlety may affect the performance profoundly. In particular, by shifting the distribution range of $r$ merely by one slot (that is, $r \in [1, CW]$), substantial throughput degradation results, and the degree of degradation increases as the number of stations gets large.

## References


[1] mac-sim. http://lion.cs.uiuc.edu/∼chunyuhu/mac-sim.html.

[2] I. Aad and C. Castelluccia. Differentiation mechanisms for IEEE 802.11. In *Proc. of IEEE INFOCOM*, 2001.

[3] D. Bertsekas and R. Gallager. *Data Networks*. Prentice Hall, 1992.

[4] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE JSAC*, 18(3), Mar. 2000.

[5] F. Cali, M. Conti, and E. Gregori. Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit. *IEEE/ACM Trans. on Networking*, 8(6), Dec. 2000.

[6] M. M. Carvalho and J. J. Garcia-Luna-Aceves. Delay analysis of IEEE 802.11 in single-hop networks. In *Proc. of IEEE ICNP*, Nov. 2003.

[7] M. M. Carvalho and J. J. Garcia-Luna-Aceves. A scalable model for channel access protocols in multihop ad hoc networks. In *Proc. of ACM MOBICOM*, Sept. 2004.

[8] E.Ziouva and T.Antonakopoulos. CSMA/CA performance under high traffic conditions: Throughput and delay analysis. *Computer Communications*, 25(3), 2002.

[9] C. H. Foh and M. Zukerman. Performance analysis of the IEEE 802.11 MAC protocol. In *Proc. of the European Wireless Conf.*, 2002.

[10] Y. Ge. QoS provisioning for IEEE 802.11 MAC protocols. Ph.D Thesis, University of Ohio State, 2004.

[11] IEEE Computer Society. IEEE standard 802.11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications. The Institute of Electrical and Electronics Engineers, New York, NY, 1997.

[12] H. Kim and J. C. Hou. Improving protocol capacity with model-based frame scheduling in IEEE 802.11-operated WLANs. In *Proc. of MOBICOM*, 2003.

[13] A. Kumar, E. Altman, D. Miorandi, and M. Goyal. New insights from a fixed point analysis of single cell IEEE 802.11 WLANs. In *Proc. of IEEE INFOCOM*, 2005.

[14] J. W. Robinson and T. S. Randhawa. Saturation throughput analysis of IEEE 802.11e enhanced distributed coordination function. *IEEE JSAC*, 22(5), 2004.

[15] Y. C. Tay and K. C. Chua. A capacity analysis for the IEEE 802.11 MAC protocol. *Wireless Networks*, 7(2), 2001.

[16] H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma. Performance of reliable transport protocol over IEEE 802.11 wireless LAN: analysis and enhancement. In *Proc. of IEEE INFOCOM*, June 2002.

[17] Y. Xiao. An analysis for differentiated service in IEEE 802.11 and IEEE 802.11e wireless LANs. In *Proc. of IEEE ICDCS*, Mar. 2004.