IEICE TRANS. COMMUN., VOL.E88–B, NO.11 NOVEMBER 2005

PAPER

Improving Throughput and Fairness in WLANs through Dynamically Optimizing Backoff

Xuejun TIAN[†a)], *Member*, **Xiang CHEN**[††], *Nonmember*, **Tetsuo IDEGUCHI**[†], *Member*, *and* **Yuguang FANG**[††], *Nonmember*

SUMMARY Given the limited channel capacity in wireless LANs, it is important to achieve high throughput and good fairness through medium access control (MAC) schemes. Although many schemes have been proposed to enhance throughput or fairness of the original IEEE 802.11 standard, they either fail to consider both throughput and fairness, or to do so with complicated algorithms. In this paper, we propose a new MAC scheme that dynamically optimizes each active node's backoff process. The key idea is to enable each node to adjust its Contention Window (CW) to approach the optimal one that will maximize the throughput. Meanwhile, when the network enters into steady state in saturated case, i.e., under heavy traffic load, all the nodes will maintain approximately identical CWs, which guarantees fair share of the channel among all nodes. A distinguishing feature of this scheme is the use of an index called average channel idle interval for optimizing the backoff process without estimating the number of active nodes in networks. We show through theoretical analysis that the average channel ideal interval can represent current network traffic load and indicate the optimal CW. Moreover, since it can be obtained through direct measurement, our scheme eliminates the need for complicated estimation of the number of active nodes as required in previous schemes, which makes our schemes simpler and more reliable when network traffic changes frequently. Through simulation comparison with previous schemes, we show that our scheme can greatly improve the throughput no matter the network is in saturated or non-saturated case, while maintaining good fairness.

key words: *WLAN, MAC, backoff, contention window, fairness*

1. Introduction

Wireless local area networks (WLANs) have become increasingly popular and widely deployed in recent years. Currently, the IEEE 802.11 MAC [1]–[3] standard includes two channel access methods: a mandatory contention-based one called Distributed Coordination Function (DCF) and an optional centralized one called Point Coordination Function (PCF). Due to its inherent simplicity and flexibility, the DCF mode is preferred and has attracted most research attention. At the same time, PCF is not supported in most current wireless cards and may result in poor performance when working alone or together with DCF, as shown in [17], [23]. In this paper, we focus on DCF.

Since all the nodes share a common wireless channel with limited bandwidth in the WLAN, it is highly desirable that an efficient and fair medium access control (MAC) scheme is employed. However, for the 802.11 DCF, there is much room for improvement in terms of both efficiency and fairness. Cali et al. pointed out in [8] that depending on the network configuration, DCF may deliver a much lower throughput compared to the theoretical throughput limit. Meanwhile, as demonstrated in [5], the fairness as well as throughput of the IEEE 802.11 DCF could significantly deteriorate when the number of nodes increases.

Although extensive research has been conducted to improve throughput [4], [6]–[12] or fairness [8], [15], except in [9], these two performance indexes are rarely considered together (more comments on these works are given in Sect. 2.2). In this paper, we aim to enhance both throughput and fairness for DCF at the same time by proposing a novel MAC scheme that Dynamically Optimizes the Backoff process. Henceforth, we call it DOB. Compared to the original 802.11 DCF and previous enhancement approaches, this scheme has the following distinguishing features:

- Unlike [8] that relies on the complicated on-line estimation of the number of active nodes, we use a simple and accurate measure called *average idle interval*, which is easily obtained and reflects network traffic load, to dynamically optimize the backoff with more sophistically algorithm. In [8], though *average idle interval* is proposed to be observed, it is used for estimating the number of active nodes in a network instead of optimizing CW directly and the number of active nodes has a complicated relation to *average idle interval*.
- It is known that in the 802.11 DCF, each node exponentially increases its contention window (CW) in case of collisions and resets it after successful transmissions. Although this is designed to avoid collisions, the fact that the CWs change drastically lead to neither fast collision resolution nor high throughput [9]. In contrast, DOB enables each node to keep a quasi-stable CW that oscillates around an *optimal* value that leads to a throughput close to the maximum. More specifically, the current CW will be decreased if it is greater than the optimal CW and be increased otherwise.
- Since each node in the network maintains its CW around the optimal value, all nodes will have equal opportunities to seize the channel. As a result, the fairness is improved compared to the original DCF.

The remainder of this paper is organized as follows. In Sect. 2, we describe the IEEE 802.11 MAC protocol and

---

Manuscript received February 3, 2005.
Manuscript revised May 19, 2005.

[†] The authors are with the Department of Information Systems, the Faculty of Information Science and Technology, Aichi Prefectural University, Aichi-ken, 480-1198 Japan.

[††] The authors are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, U.S.A.

a) E-mail: tan@ist.aichi-pu.ac.jp

DOI: 10.1093/ietcom/e88–b.11.4328

then discuss the related work. We elaborate on our key idea and the theoretical analysis for improvement in Sect. 3. Then, we present in detail our proposed DOB scheme in Sect. 4. Section 5 gives performance evaluation and the discussions on the simulation results. Finally, concluding remarks are given in Sect. 6.

2. Preliminaries

To better understand our scheme, in this section we first briefly introduce the IEEE 802.11 DCF, a distributed contention-based medium access control protocol. Then, we discuss the related work. Especially, we focus on Cali's work [8] and FCR (Fast Collision Resolution, [9]), as these two schemes resolve MAC collisions through dynamically adjusting the contention window.

2.1 Operations of the IEEE 802.11 MAC

The IEEE 802.11 DCF is based on a mechanism called carrier sense multiple access with collision avoidance (CSMA/CA). As shown in Fig. 1, in DCF mode, a node with a packet to transmit initializes a backoff timer with a random value selected uniformly from the range $[0, CW - 1]$, where CW is the contention window in terms of time slots. After a node senses that the channel is idle for an interval called DIFS (DCF interframe space), it begins to decrease the backoff timer by one for each idle time slot. When the channel becomes busy due to other nodes' transmissions, the node freezes its backoff timer until the channel is sensed idle for another DIFS. When the backoff timer reaches zero, the node begins to transmit. If the transmission is successful, the receiver sends back an acknowledgment (ACK) after an interval called SIFS (short inter-frame space). Then, the transmitter resets its CW to CW_{min}. In case of collisions, the transmitter fail to receive the ACK from its intended receiver within a specified period, it doubles its CW subject to a maximum value CW_{max}, chooses a new backoff timer, and start the above process again. When the transmission of a packet fails for a maximum number of times, the packet is dropped.

To improve channel efficiency for long packet transmissions, the IEEE 802.11 protocol can also use a short Request To Send (RTS) control frame and a short Clear To Send (CTS) frame to reserve access to the channel. In this case, collisions only happen to the transmission of RTS. Since the length of RTS is fixed, this case can be treated as a special case of where variable data packet size exists. Furthermore, if the probability that the transmission of RTS is successful is known, the throughput for different data packet sizes can be readily derived. Hereafter, we thus mainly focus on MAC protocol with comparatively short packet size without the RTS/CTS handshake.

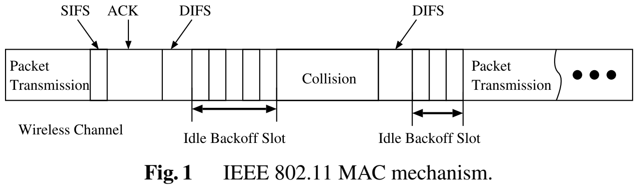


Fig. 1 IEEE 802.11 MAC mechanism.

2.2 Related Work

Considerable research efforts have been expended on either theoretical analysis or throughput improvement [4]–[12], [20]. In [5], Bianchi used a Markov chain to model the binary exponential backoff procedure. By assuming the collision probability of each node's transmission is constant and independent of the number of retransmission, he derived the saturated throughput for the IEEE 802.11 DCF. In [4], Bharghvan analyzed and improved the performance of the IEEE 802.11 MAC. Although the contention information appended to the transmitted packets can help in collision resolution, its transmission increases the traffic load and the delay results in insensitivity to the traffic changes. Kim and Hou developed a model-based frame scheduling algorithm to improve the protocol capacity of the 802.11 [20]. In this scheme, each node sets its backoff timer in the same way as in the IEEE 802.11; however, when the backoff timer reaches zero, it waits for an additional amount of time before accessing the medium. Though this scheme improves the efficiency of medium access, the calculation of the additional time is complicated since the number of active nodes must be accurately estimated.

Cali et al. [8] studied the 802.11 protocol capacity by using a p-persistent backoff strategy to approximate the original backoff in the protocol. In addition, they showed that given a certain number of active nodes and average frame length, there exists an average contention window that maximizes throughput. Based on this analysis, they proposed a dynamic backoff tuning algorithm to approach the maximum throughput. It is important to note that the performance of the tuning algorithm depends largely on the accurate estimation of the number of active nodes. However, in practice, there is no simple and effective run-time estimation algorithm due to the distributed nature of the IEEE 802.11 DCF. And proposed algorithm cannot guarantee that every node has the same CW, which may result in a poor fairness. This is because that there is no control node for DCF, every node tunes its CW according to its own view of situation of network, and after some nodes changed their CWs, which leads to changes of network traffic, the other nodes may tune with different values of CW. Meanwhile, a complicated algorithm [8], [13] would impose a significant computation burden on each node and be insensitive to the changes in traffic load.

In our previous study, a fast collision resolution (FCR) MAC algorithm was proposed [9], which actively redistributes the backoff timers for all competing nodes. Specifically, not only do transmitting nodes increase their CW in case of collisions, but also nodes in the backoff process increase their CWs and reset their backoff timer when detecting a new channel busy period. When a node successfully finishes a transmission, it restores its CW as a small contention window. To reduce the waste of time, unlike IEEE 802.11, FCR decreases backoff timers much faster (i.e., exponentially) in the case that the number of consecutive idle slots exceeds a certain threshold. In this way, FCR allows a successful node to transmit again with a high probability and thus enhances channel efficiency. The downside of FCR is that it may excessively increases CW in order to avoid collision, resulting in some wasted time slots under low-to-medium network traffic load. Moreover, FCR alone could adversely affect fairness, thereby necessitating an additional fair scheduling mechanism as shown in [9].

Fairness is another important issue in MAC protocol design for WLANs [14]. In a shared channel wireless network, throughput and fairness essentially conflict with each other as shown in [22]. The analysis in [5] demonstrated that the fairness as well as the throughput of IEEE 802.11 DCF could significantly deteriorate when the number of nodes increases. Several research works addressed this issue [8], [15], [21]. In [8], the number of active nodes needs to be estimated as mentioned and in [15], only initial contention window is adjusted and thus the contention window is not optimized.

As will be shown later, our proposed DOB preserves the advantages and overcomes the deficiencies of the work [8] and FCR. While relying on dynamic tuning of CW, it avoids complicated estimating of the number of active nodes, as is the case in [8]. Compared to the original IEEE 802.11 or FCR, since each node, without initializing its CW with the minimum value, keeps its CW close to the same optimal value, DOB can maintain fairness and keep the network operating with less fluctuation. Consequently, the network always works in a quasi-stable state. In other words, the nodes with a smaller CW than the optimal CW will increase CW and the nodes with a greater CW will decrease CW.

3. Design Motivation and Analysis

3.1 Motivation

In the IEEE 802.11 MAC, an appropriate CW is the key to providing throughput and fairness. A small CW results in high collision probability, whereas a large CW results in wasted idle time slots. In [8], Cali et al. showed that given the number of active nodes, there exists an optimal CW that leads to the theoretical throughput limit and when the number of active nodes changes, so does this optimal CW. Since in practice, the number of active nodes always changes, to let each node attain and keep using the corresponding optimal CW requires the estimation of the number of active node. However, this is not an easy task in the network environment where a contention-based MAC protocol is used. To get around this difficulty, we are thus motivated to find other effective measures that also lead us to the optimal CW and hence the maximal throughput. Therefore, we focuses on the average idle interval in the channel between two consecutive busy periods (due to transmissions or collisions) that each node locally observes. It has two merits. One is that without complex computation, each node can obtain the average idle interval online by observation, which is quite simple since the DCF is in fact built on the basis of physical and virtual carrier sensing mechanisms. The other merit is that, as shown below, the optimal average idle interval has a very simple relationship with the optimal CW that leads to the maximal throughput. It is important to note that since we believe that DOB will enable each node to use the optimal CW, we do not use the original DCF backoff algorithm. Details on DOB will be given in Sect. 4.

In the following, we derive the relationship between average idle interval and throughput through analysis. For the purpose of simplicity, we assume the frame length is constant. Later on, we will show that the performance is not sensitive to a variable frame length.

3.2 Analytical Study

In [8], the IEEE 802.11 DCF is analyzed based on an assumption that, in each time slot, each node contends for the medium with the same probability p subject to $p = 1/(E[B]+1)$, where $E[B]$ is the average backoff timer and equals $(E[CW]-1)/2$ for DCF. Since our DOB would enable all the nodes to settle on a quasi-stable CW shortly after the network is put into operation, for simplicity we assume that all the nodes use the same and fixed CW. Consequently, we have

$$p = \frac{2}{CW+1} \tag{1}$$

as all the expectation signs E can be removed.

Figure 2 shows the analytical model of channel (single wireless channel), which includes 3 types of events, successful transmission, collision, ilde. Suppose every node is an active one, i.e., always having packets to transmit. For every packet transmission, the initial backoff timer is uniformly selected from $[0, CW-1]$. For each virtual backoff time slot, it may be idle, busy due to a successful transmission, busy due to collision. Accordingly, we denote by t_{slt}, T_s, and T_{col} the time durations of the three types of virtual slots, respectively, and denote by p_{idl}, p_s, and p_{col} the associated probabilities, respectively. Thus, we can express the above probabilities as follows.

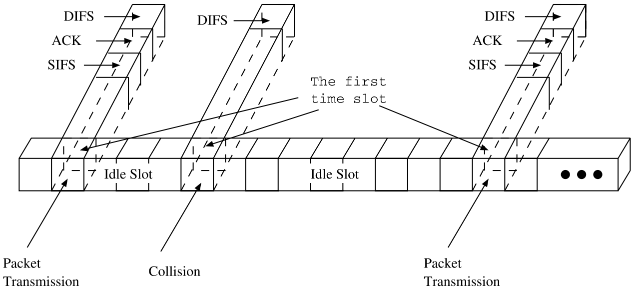


Fig. 2 Throughput vs. average idle interval.

$$
\begin{aligned}
p_{idl} &= (1-p)^n \\
p_s &= np(1-p)^{(n-1)} \\
p_{col} &= 1 - p_{idl} - p_s
\end{aligned}
\tag{2}
$$

where n is the number of active nodes. Thus, the throughput is expressed as

$$
\begin{aligned}
\rho &= \frac{Tp_s}{t_{slt}p_{idl} + T_{col}p_{cll} + T_{tx}p_s} \\
&= \frac{T}{\frac{t_{slt}p_{idl}}{p_s} + \frac{T_{col}p_{col}}{p_s} + T_{tx}}
\end{aligned}
\tag{3}
$$

where T is the transmission time of one packet, which can be obtained by subtracting overhead from $T_{tx} = T + 2\tau + SIFS + ACK + DIFS$. τ expresses the maximum propagation delay between two nodes. In the above equation, the term p_{idl}/p_s can be thought of as the average number of idle slots for every successful transmission and the term p_{col}/p_s the average number of collisions for every successful transmission. If we denote by L_{idl} the average idle interval, it can be expressed as

$$
L_{idl} = \frac{p_{idl}}{1 - p_{idl}}
\tag{4}
$$

With Eq. (1) and (2), this equation can be further written as

$$
\begin{aligned}
L_{idl} &= \frac{1}{(1 + 2/(CW-1))^n - 1} \\
&= \frac{1}{n\frac{2}{CW-1} + \ldots + \binom{n}{i}\left(\frac{2}{CW-1}\right)^{(n-i)} + \ldots + \left(\frac{2}{CW-1}\right)^n}
\end{aligned}
\tag{5}
$$

In Eq. (5), we can see that when CW is large enough, $L_{idl} = (CW-1)/(2n)$. As a matter of fact, this is the case when the network traffic load is heavy. In this case, to effectively avoid collisions, the optimal CW is large enough for the approximation $L_{idl} = (CW-1)/(2n)$ in our DOB, which is also verified through simulations.

With Eqs. (2), (3), and (5), we can express the throughput as a function of L_{idl} with SIFS=10 μs, DIFS=50 μs, ACK=304 bit, data rate=1 Mbps and time slot=20 μs, as shown in Fig. 3. Several important observations are made. First, we find that every curve follows the same pattern; namely, as the average idle interval L_{idl} increases, the throughput first rises quickly, and then decreases relatively slowly after reaching its peak. Second, although the optimal value of L_{idl} that maximizes throughput is different in cases of different frame lengths, it varies in a very small range, which hereafter is called the optimal range of L_{idl} corresponding to different framelengths. Finally, this optimal value is almost independent of the number of active nodes. Therefore, L_{idl} is a suitable measure that indicates the network throughput.

Figure 4 shows the relationship between L_{idl} and the contention window CW, as revealed in Eq. (5). It can be observed that L_{idl} is almost a linear function of CW when CW is larger than a certain value. Specifically, in the optimal range of L_{idl}, say $L_{idl} = [3, 8]$, we can estimate L_{idl} using the following linear approximation:

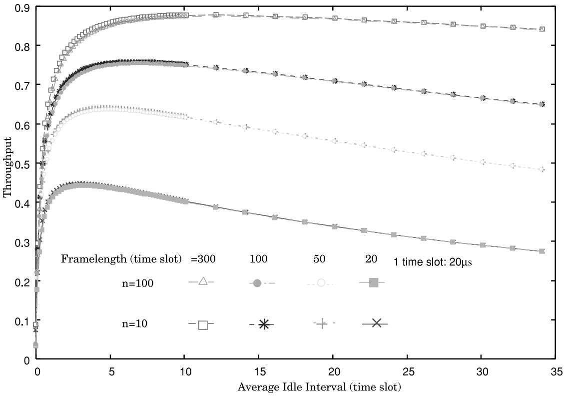


Fig. 3 Throughput vs. average idle interval.

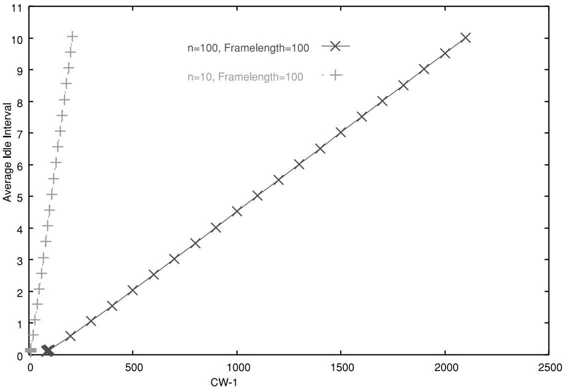


Fig. 4 Average idle interval vs. contention window.

$$
L_{idl} = \frac{CW-1}{2n} - \alpha
\tag{6}
$$

where α is a constant. Since we are interested in tuning the network to work with maximal throughput, given the nice linear relationship, we can achieve this goal by adjusting the size of CW. In other words, each node can observe the average idle interval locally and adjust its backoff window accordingly such that the network throughput is maximized.

Clearly, the above results hold when all nodes have the same CW. In reality, different nodes may have different CWs that fluctuates around the optimal CW. Next, we prove that given the average idle interval, i.e., given the idle probability p_{idl}, the achieved throughput is no less than the throughput obtained under the condition that all nodes have the same CW.

Theorem 1: Given the probability that a slot is idle is P and the number of active nodes n, the throughput is minimal in the case of $p_1 = p_2 = \ldots = p_i = \ldots = p_n$. Due to the fact that $CW = \frac{2}{p_i} - 1$, it follows that each node has the same CW.

Proof: The proof is done by mathematical induction. We first prove the case of $n = 2$. In this case, it is found that the throughput reaches minimum in case of $p_1 = p_2 =$

$1-P^{\frac{1}{2}}$. Next, we assume Theorem 1 holds when the number of active nodes is n, then we just need to prove Theorem 1 holds when the node number is $n+1$.

Divide the active nodes into two sets: one includes only one node, namely node $n+1$, and the other set includes the remaining n nodes. For the set of n nodes, in any time slot, assume the success probability is S_n, and the idle probability I_n. We have

$$S_{n+1} = p_{n+1} I_n + (1 - p_{n+1}) S_n \tag{7}$$

since $P = (1 - p_{n+1}) I_n$, we rewrite the above formula as

$$S_{n+1} = p_{n+1} \frac{P}{1 - p_{n+1}} + (1 - p_{n+1}) S_n \tag{8}$$

We can find from the above equation that only when S_n reaches minimum the S_{n+1} reaches the minimum. The condition that S_n reaches minimum is $p_1 = p_2 = ... = p_n = p$, then $S_n = np(1-p)^{n-1}$, $I_n = (1-p)^n$, $P = (1-p_{n+1})(1-p)^n$ and $p = 1 - (\frac{P}{1-p_{n+1}})^{\frac{1}{n}}$, we have

$$S_{n+1} = p_{n+1} \frac{P}{1 - p_{n+1}} + nP \left[\left(\frac{P}{1 - p_{n+1}} \right)^{-\frac{1}{n}} - 1 \right] \tag{9}$$

Take the derivative of S_{n+1} with respect to p_{n+1} and let it equal 0, i.e., let $S'_{n+1} = 0$, we obtain $p_{n+1} = 1 - P^{\frac{1}{n+1}}$ and also $p = 1 - P^{\frac{1}{n+1}}$. Theorem 1 is proved.

Theorem 1 reveals two important points. First, if we keep the average idle interval as the one corresponding to the peak throughput shown in Fig. 3, the achieved throughput will not be less than the peak throughput, as the peak throughput is derived under the condition that all nodes have the same CW. Second, it shows that there is a tradeoff between throughput and fairness. If all nodes keep the same CW, which means the channel is fairly shared among all nodes, the achieved throughput is sacrificed.

By detecting the average idle interval, each node can adjust its current CW around the optimal CW at runtime. Assume the observed current average idle interval is l_{idl}, the optimal CW is CW_o, and the corresponding optimal average channel idle interval is L_{io}, given Eq. (6), we can estimate CW_o as

$$CW_o = (CW_c - 1) \frac{L_{io} + \alpha}{l_{idl} + \alpha} + 1 \tag{10}$$

where CW_c is the current CW. Clearly, we obtain the optimal CW while avoiding the difficult task of estimating the number of active nodes. Then, we can adjust the current CW based on Eq. (10) so as to approach the optimal CW and hence tune the network to deliver high throughput. In the following, we give the tuning algorithm in detail.

4. DOB Scheme

In this section, we describe the DOB scheme in which each node adapts its backoff process according to the observed average idle interval l_{idl}, which reflects the network traffic load. The goal is that each node always uses a contention window close to the optimal CW.

Unlike the IEEE 802.11, DOB divides a node's backoff process into three stages, namely stage 0, 1, and 2. When an arbitrary node starts backoff, depending on the initially chosen backoff timer (BT), it may enter stage 1 or 2. After a collision, it may enter into stage 0. At each stage, the node decrements its backoff timer when an idle time slot is detected. It starts transmission only when its BT reaches 0 and it is at stage 1 or 2. At the end of stage 0 and stage 1, it refreshes its CW based on the average idle interval l_{idl} observed in previous stages according to Eq. (14), a revised version of Eq. (10), so that a high aggregate throughput can be achieved. We introduce two parameters K_h and K_l as thresholds for the observed channel idle interval l_{idl}. Specifically, K_h is the threshold corresponding to high traffic load and K_l the threshold corresponding to low traffic load. When observed l_{idl} is lower than K_h, the node increases its CW; when l_{idl} is larger than K_l, the node decreases its CW. K_h and K_l are defined as follows:

$$K_h = K'_h - \frac{CW_i - 1}{CW_{ct}} \tag{11}$$

$$K_l = K'_l - \frac{CW_i - 1}{CW_{ct}} \tag{12}$$

where the range defined by $[K'_h, K'_l]$ corresponds to the optimal range around L_{io}, CW_i is the current CW of node i, and CW_{ct} is a constant that is the same for each node.

Ideally, each node should have the same CW when the network enters into steady state in saturated case; in reality, however, this is not the case. When a new active node initializes its CW as CW_{min}, it may be different from the CWs used in other nodes; or when traffic load changes, CWs of different nodes may change accordingly and differ from each other in a short term. As a result, fairness is impaired. We thus introduce the term $(CW_i - 1)/CW_{ct}$ to enhance the short-term fairness, as noticed in Eqs. (11) and (12). Its role can be better understood as follows. Since $(CW_i - 1)/CW_{ct}$ is independent of the average idle interval, it forces nodes with larger CWs to decrease their CW and nodes with smaller CWs to increase their CW. More specifically, when a node obtains the average idle interval as l_{idl}, it will compare it with K_h and K_l and then adapt its CW according to the algorithm presented below. When $l_{idl} < K_h$, it will increase its CW, and when $l_{idl} > K_l$, it will decrease its CW. According to Eq. (11), it can be seen that the smaller the CW of a node, the larger K_h because of the term $(CW_i - 1)/CW_{ct}$ and hence the higher probability that $l_{idl} < K_h$. This leads to the higher probability for the node to increase its CW. On the other hand, according to Eq. (12), the larger the CW of a node, the smaller K_l and hence the higher probability that $l_{idl} > K_l$. This leads to the higher probability for the node to decrease its CW. In this way, the CW of each node always converges to the same value even in a short term.

For L_{io} in Eq. (10), we also introduce the term $(CW_i - 1)/CW_{ct}$ and define L_c as a revised version of L_{io}:

$$L_c = L_{io} - \frac{CW_i - 1}{CW_{ct}} \tag{13}$$

Replacing L_{io} in Eq. (10) with L_c, we have

$$CW_o = (CW_c - 1)\frac{L_c + 0.5}{l_{idl} + 0.5} + 1 \tag{14}$$

where we approximate α in Eq. (10) with 0.5. Obviously, we have $K_h^{'} < L_{io} < K_l^{'}$ and $K_h < L_c < K_l$. Since each node adjusts its CW to approach the same value, we assume that finally every node has almost the same and quasi-stable CW in steady state, which means the observed average idle interval l_{idl} is in the range $K_h < L_c < K_l$. In the steady state, we thus can assume $l_{idl} = L_c$. Combining Eqs. (6) and (13), in which $CW_i = CW$, we obtain the value of the quasi-stable CW as

$$CW = \frac{L_{io} + 0.5}{\frac{1}{2n} + \frac{1}{CW_{ct}}} + 1 \tag{15}$$

As a matter of fact, it is also verified by the average CW obtained from the simulations in Sect. 5.2. Note that the CW in Eq. (15) is slightly different from the optimal value in Eq. (6) because of the introduction of CW_{ct}, which is a tradeoff between fairness and aggregate throughput. When the maximum number of nodes is less than or equal to 100, we can set $CW_{ct} = 250$ as shown in our simulation. When the number of active nodes in the network is more than 100, as far as throughput is concerned, we can set a larger CW_{ct} to keep the CW in Eq. (15) closer to the optimal value in order to increase throughput ; however, this could lead to a little degradation in fairness as each node adapts its CW in a slower pace.

To avoid measuring the average channel idle interval in a short term, a node calculates the average idle interval only if has observed at least a certain number of idle slots, with the number specified by Observation Window (OW). DOB adopts different backoff processes for new transmissions and retransmissions after a collision. In the case of new transmission, a node, say node A, follows the following algorithm (Algorithm I):

1. Node A uses its current CW if it has one; otherwise it selects CW_{min} as its current CW and set its backoff timer (BT) to $uniform[0, CW - 1]$. Node A enters into backoff stage 2 if $BT < OW$ or stage 1 otherwise.

2. Node A in stage 2 decreases its BT by 1 whenever it detects an idle time slot. If $BT = 0$, it begins transmission. If node A is in stage 1, it counts the number of consecutive idle slots while decreasing BT by 1 for each idle slot. When BT reaches zero, it calculates l_{idl}. Then, it compares l_{idl} with K_l and K_h. If l_{idl} is in the range between K_h and K_l, i.e., $K_h < l_{idl} < K_l$, node A begins transmission immediately without changing CW; otherwise it acts as follows.

i) if $l_{idl} > K_l$, node A starts transmission immediately and decreases its CW as $newCW = (CW - 1)\frac{L_c+0.5}{l_{idl}+0.5} + 1$ according to Eq. (14).

ii) if $l_{idl} < K_h$, node A increases its CW as $newCW = (CW - 1)\frac{L_c+0.5}{l_{idl}+0.5} + 1$ and resets its BT as $BT = uniform[0, newCW - CW]$. Then node A enters backoff stage 2.

In the case of collision, node A follows the following algorithm (Algorithm II):

1. Node A sets its BT as $BT = uniform[0, 2CW + 1]$ without changing its CW.

2. If $BT < OW$, node A enters into backoff stage 2 and begins decreasing BT by 1 for every idle slot until $BT = 0$; then it starts transmission. If $BT \geq OW$, node A enters into backoff stage 0; while decreasing BT by 1 for each idle slot, it starts the calculation process of l_{idl} as stated previously. After OW idle slots, node A calculates l_{idl}. If l_{idl} is in the range between K_h and K_l, i.e., $K_h < l_{idl} < K_l$, node A enters into backoff stage 2 without any other adjustment. Otherwise, node A acts as follows.

i) if $l_{idl} > K_l$, node A still uses the current CW and resets BT as $BT = uniform[0, CW - 1]$.

ii) if $l_{idl} < K_h$, node A increases its CW as $newCW = (CW - 1)\frac{L_c+0.5}{l_{idl}+0.5} + 1$ and resets its BT as $BT = uniform[0, newCW - 1]$.

Then node A enters into stage 1 and follows step 2 in Algorithm I.

5. Performance Evaluation

In this section, we focus on evaluating the performance of our DOB through simulations, which are carried out on OPNET Modeler [24]. For comparison purpose, we also present the simulation results for the IEEE 802.11 DCF and FCR. In all the simulations, we consider the basic MAC scheme. In other words, the RTS/CTS mechanism is not used. The DCF-related parameters are shown in Table 1 and the DOB-specific parameters in Table 2. Though simulation parameters are set on the basis of IEEE 802.11b and the bit rate is lower than IEEE 802.11 a/g/e, DOB will also work with them. We will discuss this item in the next section. Suppose that nodes can distinguish collision and failure resulted from error, then they can ignore the case of error to

Table 1 Network configuration.

Parameter	Value
$MinCW$	16
$MaxCW$	1024
SIFS	10 μsec
DIFS	50 μsec
Slot Length	20 μsec
aPreamblLength	144 bits
aPLCPHeaderLength	48 bits
Bit rate	1 Mbps

Table 2 Backoff parameters.

Parameter	Value
$K_h^{'}$	5.8
$K_l^{'}$	6.0
L_{io}	5.9
OW	15
CW_{ct}	250

adjust their CWs. So, we assume the channel without error.

We assume each node generates traffic according to a Poisson process with the same arrival rate. The arrival rate is kept increasing until the network is saturated. As shown below, DOB exhibits a better performance than the IEEE 802.11 and FCR in terms of throughput and fairness.

5.1 Throughput

Firstly, we present the throughput obtained for the three schemes, i.e., DOB, FCR and the IEEE 802.11, under different offered load. Figures 5, 6, 7, 8, 9, 10 show the throughput results when the number of nodes is 10, 50, and 100, and the packet sizes are 2048 bits and 5120 bits, respectively. Note that the packet size is the size of payload data and does not include MAC overhead, which is the reason that the simulation results are lower than the theoretical values.

It can be observed that when the traffic load is low, say lower than 0.6, the throughputs of DOB and the IEEE 802.11 are almost the same and equal to the offered load, whereas the throughput of FCR is lower. It can be explained as follows. For both DOB and the 802.11, since the offered

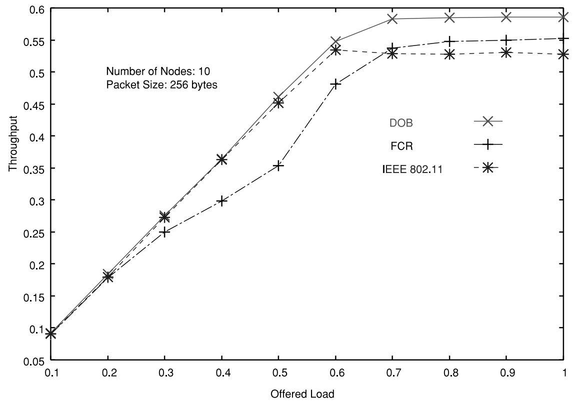


Fig. 5 Throughput vs. offered traffic with 10 nodes.

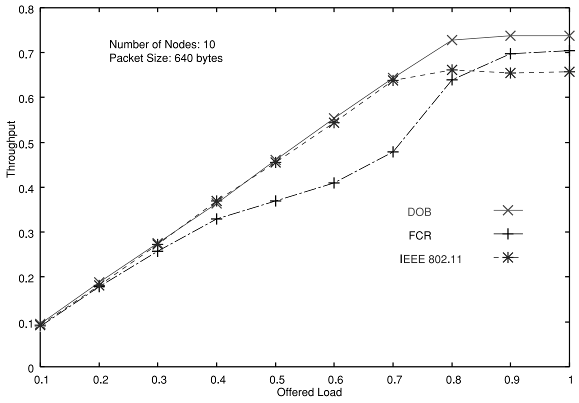


Fig. 6 Throughput vs. offered traffic with 10 nodes.

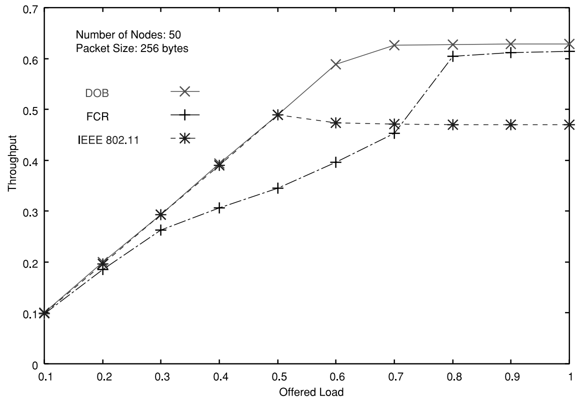


Fig. 7 Throughput vs. offered traffic with 50 nodes.

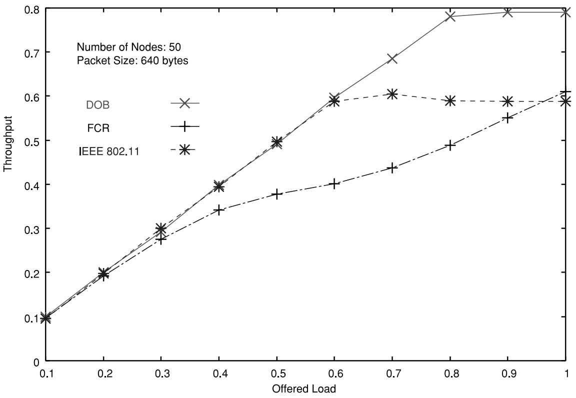


Fig. 8 Throughput vs. offered traffic with 50 nodes.

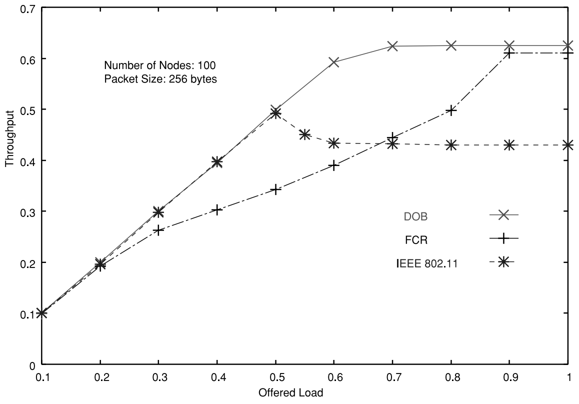


Fig. 9 Throughput vs. offered traffic with 100 nodes.

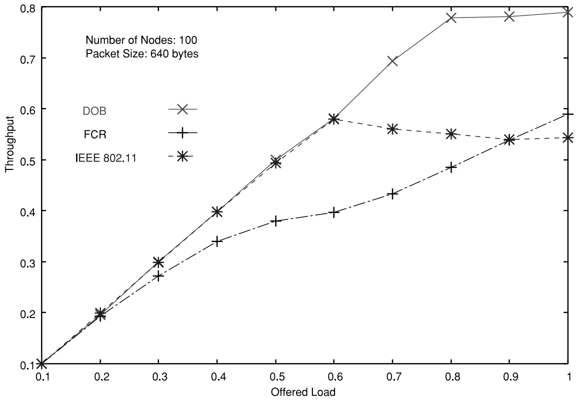


Fig. 10 Throughput vs. offered traffic with 100 nodes.

load is low, the MAC collisions are slight and all the offered traffic can get through. Conversely, each node in FCR always aggressively increases its CW even when collisions are not severe. As a result, it leads to wasted idle slots and hence inefficiency in the case of nonsaturation. This observation is more pronounced as the number of nodes increases. When the traffic load becomes heavy and the network enters into saturation, we see that the throughput of the 802.11 first increases with the traffic load, then slightly decreases after reaching the peak, and finally stabilizes at a certain value. This phenomenon is due to the fact that the maximum throughput of the 802.11 is larger than its saturated throughput. For both DOB and FCR, their throughputs first increase with the traffic load and then becomes stable. In the stable state, it can be seen that the 802.11 yields the lowest throughput among the three schemes. FCR is much better than the 802.11 because it can resolve the collisions in saturated case faster and more efficiently; and when a node seizes the channel, it will continuously transmit with a very high probability, resulting in high channel utilization. Even so, our DOB outperforms FCR, though the difference in the throughput dwindles as the number of nodes become large.

We also compare the throughputs of these three schemes as a function of the average packet size in saturated case, where each node always has packets in its buffer waiting for transmission. Figure 11, Fig. 12 and Fig. 13 present the throughputs corresponding to the cases where the number of nodes is 10, 50, and 100, respectively. In all cases, we find that the throughput increases along with the average packet size. This is because in the saturated case, given the number of nodes, the probabilities p_{idl}, p_s, and p_{col} are constant. Accordingly, according to Eq. (3), the throughput gets larger if T gets larger while the overhead is the same. We also find that in all cases, our DOB achieves the highest throughput; as the number of nodes increases, the difference between in the throughputs of DOB and FCR tends to diminish, which is consistent with Fig. 5 to Fig. 10. Comparing with theoretical results in Fig. 12 and Fig. 13, we can find that the simulation results of DOB consist well with the theoretical value in the cases of 50 nodes and 100 nodes, but there is a difference in the case of 10 nodes. Observing the simulation, we find that when one node changes its CW, the system with less nodes is affected more remarkably and CW cannot be kept close to the optimum. As a method to overcome this problem, we can introduce a smoothing factor ([25]) to avoid sharp changes with a cost of decrease in sensitivity to traffic changes.

To sum up, the throughput performance of DOB is the best among the three schemes either in non-saturated case or saturated case. This is attributed to the fact that our DOB uses the average idle interval to adapt CW to approach the optimal CW, which can efficiently resolve collisions and lead to high throughput. Compared to FCR, DOB overcomes FCR's inefficiency in non-saturated case while being slightly more efficient in dealing with collisions in saturated case.

5.2 Fairness

To evaluate the fairness of DOB, we adopt the following Fairness Index (FI) [16] that is commonly accepted:

$$FI = \frac{(\sum_i T_i/\phi_i)^2}{n\sum_i (T_i/\phi_i)^2} \tag{16}$$

where T_i is throughput of flow i, ϕ_i is the weight of flow i (normalized throughput requested by each node). Here,

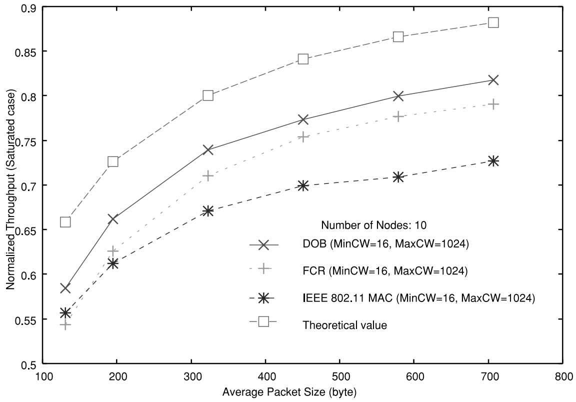


Fig. 11 Simulation results of throughput with 10 nodes.

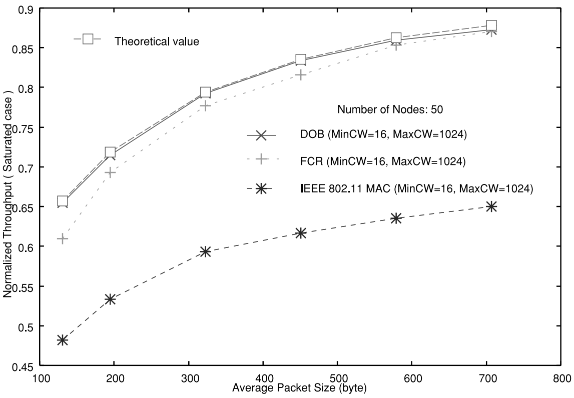


Fig. 12 Simulation results of throughput with 50 nodes.

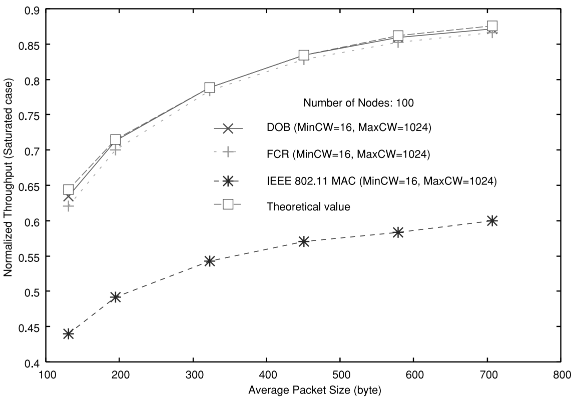


Fig. 13 Simulation results of throughput with 100 nodes.

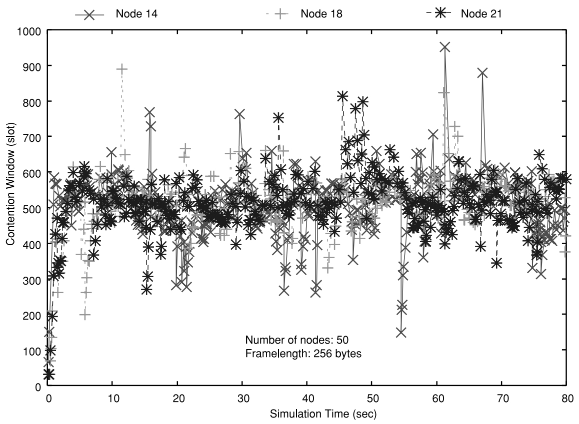


Fig. 14 Changes of CW in simulation with 50 nodes.

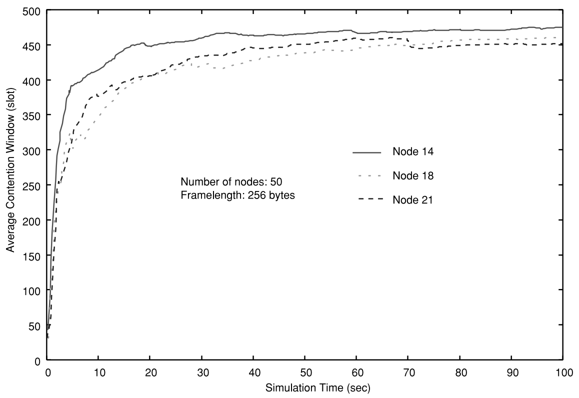


Fig. 15 Average CW in simulation with 50 nodes.

we assume all nodes have the same weight in simulation. According to Eq. (16), $FI \leq 1$, where the equation holds only when all T_i/ϕ_i are equal. Normally, a higher FI means a better fairness.

Before comparing the fairness indexes between DOB and the IEEE 802.11[†], we show how each node's CW changes in the course of simulation. Figure 14 shows the instantaneous change of the CW for three nodes that are randomly selected from a tatol of 50 nodes. We see that while the CWs fluctuate from time to time, they have close average values, which are illustrated in Fig. 15. In Fig. 15, the average CW is about 450, close to the value 464.286 obtained by Eq. (15). As the number of nodes increases, say 100, we have similar results as shown in Figs. 16 and 17.

The fairness index of DOB and the IEEE 802.11 are shown in Fig. 18. It can be observed that the fairness of DOB within 10s and 20s periods are significantly improved over that of the IEEE 802.11. It can also be seen that as the number of nodes rises, the fairness drops quickly for the 802.11, whereas for DOB, the fairness only slightly decreases. This is because DOB ensures that all the nodes use about the same CW that is around the optimal value, as shown in Figs. 14, 15, 16 and 17. Therefore, it can achieve a better fairness.

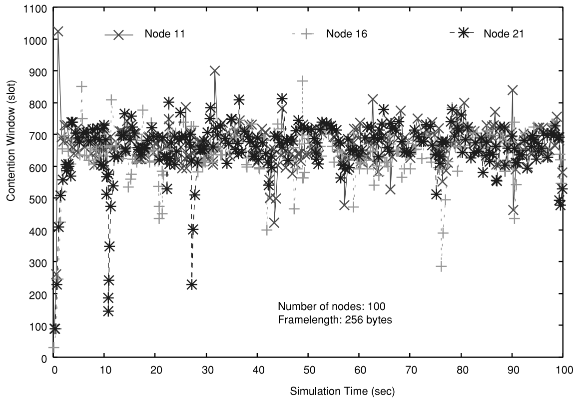


Fig. 16 Changes of CW in simulation with 100 nodes.

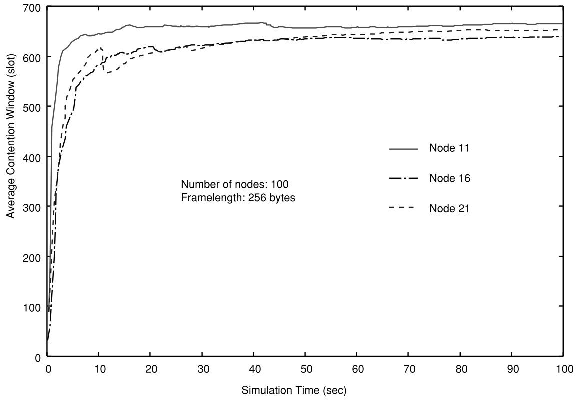


Fig. 17 Average CW in simulation with 100 nodes.

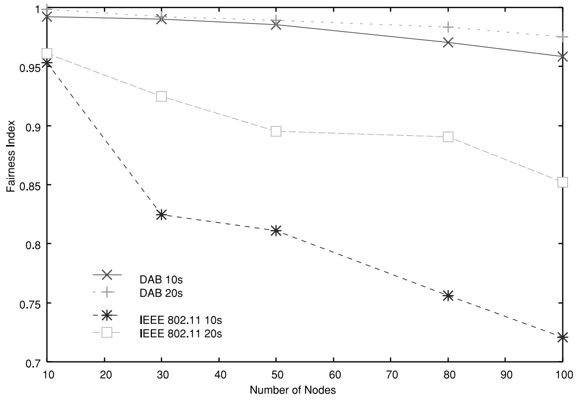


Fig. 18 Fairness index.

5.3 Discussion

The results of simulation based on IEEE 802.11b show DOB works well. Generally, DOB will also work with IEEE 802.11 a/g. The main difference of 802.11a/b/g is

---

[†] Note we do not include FCR since it depends on an additional scheduling algorithm to achieve good fairness. As a matter of fact, FCR itself negatively affects fairness as whichever node seizes the channel always has a high probability of seizing the channel again.

in the PHY layer. And they face the common problem of contention resolution for distributed contention-based MAC protocols.

Since higher data transmission rate just means shorter transmission time for the frame with a certain framelength, it is reasonable to think that the DOB will achieve better outcomes with 11a/g by dynamically changing threshold of optimal value of idle length corresponding to different proportion of overhead and payload in the cases of different data transmission rates. As for the 802.11e, it supports QoS in the way of allowing request with high priority to contend media with a short backoff time parameter. But, in this way, there is a problem remained that backoff algorithm will not work effectively if the number of nodes with high priority increases over a certain value and the aggregate throughput will decrease dramatically. DOB can be utilized to improve IEEE 802.11e. It is an interesting subject to adopt DOB to achieve both of high throughput and good QoS, for example, setting different idle lengths for different priorities, and it should be carried out continually.

When nodes with DOB coexist with legacy devices without DOB, DOB has to cooperate with them. In the case of low traffic load, nodes with DOB keep their CWs smaller than that of nodes without DOB, then nodes with DOB will be prior to those without DOB to access media and nodes without DOB almost have no dramatic difference. In this case, higher aggregate throughput can be achieved. On the other hands, in the case of high traffic load, nodes with DOB keep big CWs awaring of high traffic load. In this case, nodes without DOB will be prior to access to the media and make aggregate throughput lower. One way to alleviate the problem is that nodes with DOB decrease their CWs on purposes. Certainly, this will lower the effect of enhancement of aggregate throughput.

6. Conclusion

In this paper, we first show that under the assumption that all the nodes use the same and fixed contention window, an index called average channel idle interval can approximately indicate network traffic and has a simple relationship with the optimal contention window CW that leads to the maximal throughput. DOB is more meaningful for RTS/CTS network since RTS and CTS can be thought as packets with fixed length and DOB will not be affected by packet size. In DOB, nodes just need to confirm if media is busy or idle to obtain *average idle interval* without distinguishing between successful transmission and collision which is difficult in some cases. Meanwhile, if all the nodes use the same CW, they will fairly share the common wireless channel. We also prove that if each node uses different CW, the obtained throughput will not be less than the maximal throughput mentioned above. Based on the analysis, we propose a MAC scheme called DOB to approach this ideal case. In DOB, each node dynamically adjusts its backoff process according to the observed channel average idle interval. The goal is to approach the optimal CW and thus achieve high throughput. At the same time, when the network becomes steady, all the nodes will use a CW that oscillates around the same optimal value, leading to good fairness.

Through both analysis and simulation, our scheme has the following advantages. First, as shown in the analysis, the average idle interval of channel is a suitable index for each node to grasp the network traffic situation and is insensitive to the change in packet length or the number of active nodes. Each node only needs to adjust its backoff process based on the observed average channel idle interval, avoiding the difficult task of estimating the number of active nodes in a changing network environment as required in [8]. Second, compared with the original IEEE 802.11, DOB achieves much higher throughput in saturated case; compared with FCR, DOB overcomes its inefficiency in non-saturated case. This is attributed to the fact that in DOB, each node always adjust its backoff to approach the optimal CW and thus leads to high throughput. Finally, DOB achieves good fairness since when the network becomes stable, all the nodes maintain almost identical average CW. Farthermore, DOB can be used to improve IEEE 802.11e and we will continue to work on it.

Acknowledgment

This research was partially supported by the Ministry of Education, Culture, Sports, Science and Technology, Grant-in-Aid for Exploratory Research, 17650020, 2005.

References

[1] IEEE Std. 802.11-1999, Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, Reference number ISO/IEC 8802-11,1999(E), IEEE Std. 802.11, 1999 ed., 1999.

[2] B.P. Crow, I. Widjaja, J.G. Kim, and P.T. Sakai, "IEEE 802.11 wireless local area networks," IEEE Commun. Mag., vol.35, pp.116–126, Sept. 1977.

[3] C. Fullmer and J. Garcia-Luna-Aceves, "Floor acquisition multiple access (FAMA) for packet-ratio networks," Proc. SIGCOMM'95, pp.262–273, Cambridge, MA, 1995.

[4] V. Bharghvan, "Performance evaluation of algorithms for wireless medium access," IEEE International Computer Performance and Dependability Symposium IPDS'98, pp.86–95, 1998.

[5] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," IEEE J. Sel. Areas Commun., vol.18, no.3, pp.535–547, March 2000.

[6] Y.C. Tay and K.C. Chua, "A capacity analysis for the IEEE 802.11 MAC protocol," ACM/Baltzer Wireless Networks, vol.7, no.2, pp.159–171, March 2001.

[7] J.H. Kim and J.K. Lee, "Performance of carrier sense multiple access with collision avoidance protocols in wireless LANs," Wirel. Pers. Commun., vol.11, no.2, pp.161–183, 1999.

[8] F. Cali, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit," IEEE/ACM Trans. Netw., vol.8, no.6, pp.785–799, Dec. 2000.

[9] Y. Kwon, Y. Fang, and H. Latchman, "A novel MAC protocol with fast collision resolution for wireless LANs," IEEE INFOCOM'03, April 2003.

[10] J. Weinmiller, H. Woesner. J.P. Ebert, and A. Wolisz, "Analyzing and tuning the distributed coordination function in the IEEE 802.11 DFWMAC draft standard," Proc. MASCOT, San Jose, CA, Feb. 1996.

[11] 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," IEEE INFOCOM'02, vol.2, pp.599–607, June 2002.

[12] H.S. Chhaya and S. Gupta, "Performance modeling of asynchronous data transfer methods of IEEE 802.11 MAC protocol," ACM/Baltzer Wireless Networks, vol.3, pp.217–234, 1997.

[13] G. Bianchi and I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network," IEEE INFOCOM'03, vol.2, pp.844–852, April 2003.

[14] T. Ozugur, M. Naghshineh, P. Kermani, and J.A. Copeland, "Fair media access for wireless LANs," IEEE GLOBECOM'99, pp.570–579, 1999.

[15] P. Yong, H. Wu, S. Cheng, and K. Long, "A new self-adapt DCF algorithm," IEEE GLOBECOM'02, vol.1, pp.87–91, Nov. 2002.

[16] R. Jain, A. Durresi, and G. Babic, "Throughput fairness index: An explanation," ATM Forum/99-0045, Feb. 1999.

[17] M.A. Visser and M.E. Zarki, "Voice and data transmission over an 802.11 wireless network," IEEE PIMRC'95, pp.648–652, 1995.

[18] A. Chandra, V. Gummalla, and J.O. Limb, "Wireless medium access control protocols," IEEE Communications Surveys, Second Quarter, vol.3, no.2, pp.2–15, 2000.

[19] D.J. Goodman, R.A. Valenzuela, K.T. Gayliard, and B. Ramamurthi, "Packet reservation multiple access for local wireless communications," IEEE Trans. Commun., vol.37, pp.885–890, Aug. 1989.

[20] H. Kim and J. Hou, "Improving protocol capacity with model-based frame scheduling in IEEE 802.11-operated WLANs," Proc. ACM MobiCom 2003, pp.190–204, Sept. 2003.

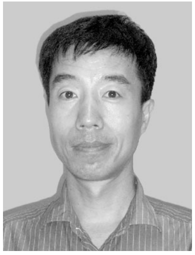
[21] N.H. Vaidya, P. Bahl, and S. Gupta, "Distributed fair scheduling in a wireless LAN," Proc. ACM MobiCom 2000, pp.167–178, 2000.

[22] T. Nandagopal, T.-E. Kim, X. Gao, and V. Bharghavan, "Achieving MAC layer fairness in wireless packet networks," Proc. ACM MobiCom 2000, pp.87–98, 2000.

[23] J.Y. Yeh and C. Chen, "Support of multimedia services with the IEEE 802-11 MAC protocol," Proc. ICC'02, pp.600–604, 2002.

[24] OPNET Modeler. http://www.opnet.com

[25] W.R. Stevens, TCP/IP Illustrated, Volume 1: The Protocols, Addison-Wesley, Reading, MA, 1994.



Xuejun Tian graduated from Hebei University, China in 1985. He received his MS degree from Department of Electrical and Mechanical Engineering, Tianjin Institute of Technology, China in 1991, and Ph.D. degree from Department of Intelligence and Computer Science, Nagoya Institute of Technology, Japan, in 1998, respectively. Since 1998, he has been an assistant professor in Department of Information Systems, Faculty of Information Science and Technology, Aichi Prefectural University, Japan. From July 2003 to June 2004, he was a Visiting Assistant Professor in Department of Electrical and Computer Engineering at University of Florida, Gainesville, FL. Dr. Tian is a member of the IEEJ (the Institute of Electrical Engineers of Japan). His research interests includes QoS, wireless networks, mobile communications and ubiquitous computing.

Xiang Chen received his Ph.D. degree in electrical and computer engineering from the University of Florida in 2005, and received his M.E. and B.E. degrees in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 2000 and 1997, respectively. His research interests include resource management, medium access control, and QoS in wireless networks. He is a student member of IEEE and a member of Tau Beta Pi.

Tetsuo Ideguchi received the B.S. degree in telecommunication engineering from the University of Electro-Communications in 1972, and the Ph.D. degree in telecommunication engineering from Tohoku University in 1993. He is a professor in the Faculty of Information Science and Technology, Aichi Prefectural University, Aichi, Japan and working on the research of network architecture, LAN, network management and mobile communications. He is a member of IEEE and IPSJ in Japan.

Yuguang Fang received the B.S. and M.S. degrees in Mathematics from Qufu Normal University, Qufu, Shandong, China, in 1984 and 1987, respectively, a Ph.D. degree from Department of Systems, Control and Industrial Engineering at Case Western Reserve University, Cleveland, Ohio, in January 1994, and a Ph.D. degree from Department of Electrical and Computer Engineering at Boston University, Massachusetts, in May 1997. From 1987 to 1988, he held research and teaching positions in both Department of Mathematics and the Institute of Automation at Qufu Normal University. He held a post-doctoral position in Department of Electrical and Computer Engineering at Boston University from June 1994 to August 1995. From June 1997 to July 1998, he was a Visiting Assistant Professor in Department of Electrical Engineering at the University of Texas at Dallas. From July 1998 to May 2000, he was an Assistant Professor in the Department of Electrical and Computer Engineering at New Jersey Institute of Technology, Newark, New Jersey. Since May 2000, he has been an Assistant Professor in the Department of Electrical and Computer Engineering at University of Florida, Gainesville, Florida. His research interests span many areas including wireless networks, mobile computing, mobile communications, automatic control, and neural networks. He has published over eighty papers in refereed professional journals and conferences. He has received the National Science Foundation Faculty Early Career Development Award in 2001 and the Office of Naval Research Young Investigator Award in 2002. Dr. Fang has actively engaged in many professional activities. He is a senior member of the IEEE and a member of the ACM. He is an Editor for IEEE Transactions on Communications, an Editor for IEEE Transactions on Wireless Communications, an Editor for ACM Wireless Networks, an Area Editor for ACM Mobile Computing and Communications Review, an Associate Editor for Wiley International Journal on Wireless Communications and Mobile Computing, and Feature Editor for Scanning the Literature in IEEE Personal Communications. He was an Editor for IEEE Journal on Selected Areas in Communications: Wireless Communications Series. He has also actively involved with many professional conferences such as ACM MobiCom'02, ACM MobiCom'01, IEEE INFOCOM'00, INFOCOM'98, IEEE WCNC'02, WCNC'00 (Technical Program Vice-Chair), WCNC'99, and International Conference on Computer Communications and Networking (IC3N'98) (Technical Program Vice-Chair).