International Journal of Multimedia and Ubiquitous Engineering
Vol. 4, No. 2, April, 2009

Optimizing Channel Access in Wireless Local Area Network Environments with a New Backoff Approach

Hao-Ming Liang[1], Sherali Zeadally[2], Naveen K Chilamkurti[3], and Ce-Kuen Shieh[1]
[1]EE Department, National Cheng Kung University, Taiwan
[2]CSIT Department, University of the District of Columbia, USA
[3]CSCE Department, La Trobe University, Australia
joeliang@hpds.ee.ncku.edu.tw; szeadally@udc.edu; n.chilamkurti@latrobe.edu.au

Abstract

Over the past few years, several backoff algorithms such as Exponential Increase Exponential Decrease (EIED) and Adaptive Enhanced Distributed Coordination Function (AEDCF)) have been proposed for wireless local area networks to improve channel access. We propose a new backoff technique that monitors the number of backoff counter pauses experienced and modifies the contention window accordingly. We evaluate and compare the performance of our proposed approach with EIED and AEDCF channel access techniques. Our simulations results, obtained under different network conditions, show improved performance for metrics such as the fairness index and end-to-end delay.

1. Introduction

With the rapid growth of wireless networks and their widespread deployment, wireless networks play an important role in data communications today. People are able to connect to the Internet via Wireless LANs (WLANs) in hot spots such as coffee shops, restaurants, or hospitals. In general, new user services (e.g., video streaming) have large amounts of data, so a large bandwidth is expected to run these services. Other services with delay sensitive requirements also include applications such as interactive multimedia, Voice over IP (VoIP) and video conferencing. These applications need to satisfy Quality of Service (QoS) requirements such as packet loss and end-to-end delay. In contrast to wired networks, the bandwidth of wireless network is limited. Furthermore, a wireless channel is error-prone and packets can be discarded in transmission due to wireless errors such as signal fading or interference. Thus, the efficiency of a wireless channel access becomes a critical issue.

IEEE 802.11 [1] is the dominant technology used in Wireless LANs. In IEEE 802.11 standard, a MAC protocol supports two coordination functions. One of them is Point Coordination Function (PCF) which provides a polling-based service and is only available in an infrastructure network mode. The Access Point (AP) acts as the administrator and determines channel access by sending a poll frame to each station in the network. The other transmission function defined in a MAC protocol is the Distributed Coordination Function (DCF) which is a contention-based service and is available both in Ad Hoc and infrastructure network modes. In DCF, the stations adopt a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism and contend for channel access. This contention behavior results in collisions among stations and therefore additional time is spent to recover from collisions. PCF is more efficient in terms of throughput performance due to contention free. However, it is not a mandatory support in IEEE 802.11standard. Thus, most research [2-

4] focuses on transmission quality and efficiency for DCF functions of the IEEE 802.11 standard.

As mentioned previously, there are unavoidable overheads such as header and Inter Frame Space (IFS) within the network. In [5], the authors describe the existence of throughput upper bound even when the data rate is high. In [6] the authors show that the efficiency of the network decreases when there are more active stations within the system. In [6], a Markov chain model is used to represent the behavior of a DCF function and a throughput formula is proposed. From the analysis in [6], the Binary Exponential Backoff (BEB) algorithm is the key factor that influences system efficiency. In 802.11 standards, the size of the Contention Window (CW) is doubled when transmission fails and reset to initial value CW_{min} once transmission is successful. This reset behavior becomes very inappropriate when numerous stations are contending within wireless channel. This can cause more collisions and decreases the whole system utilization.

In prior research efforts [7-10], the authors discuss techniques to reduce the collision rate by controlling the Contention Window (CW) size in their backoff algorithms to improve the throughput performance. The basic concept central to these methods is the estimation of the system load using the transmissions status and the appropriate CW size is computed. The size of CW could be additive or multiplicatively increased because of collisions and additive or multiplicatively decreased as a result of successful transmissions. However, the estimation method is based on partial observations, such as that each station uses its own status of transmissions to represent the whole system. The status of transmissions and system load may have a positive correlation but is not sufficient to precisely set CW value. As a result, such an imprecise calculation of CW size for stations introduces a fairness issue.

In this work, we propose a new method to estimate the number of active stations and find an appropriate value of CW within a wireless network. The countdown procedure in backoff mode is paused when other stations use the wireless channel at the same time. Therefore, each pause represents more than one station using the wireless channel and the number of pauses could give a sense of the system status. The proposed method, known as Pause Count Backoff (PCB), counts pauses during the countdown procedure and sets an appropriate CW size for the current condition. Using simulation results, we show that the proposed method improves system utilization and reduces the end-to-end delay when compared with other previous schemes such as the Distributed Coordination Function (DCF), Exponential Increase Exponential Decrease (EIED), and Adaptive Enhanced Distributed Coordination Function (AEDCF) algorithms. The fairness index of PCB is approximately 1 in most of the network simulation scenarios.

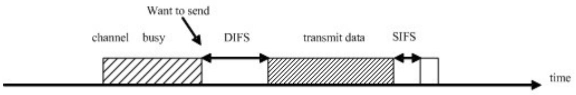
This paper is organized as follows. Section 2 describes the DCF function and discusses the overheads of transmissions in wireless channel. Brief reviews of related studies are presented in section 2. In section 3, we describe our proposed PCB algorithm. Simulation results and a performance analysis of our proposed PCB algorithm are discussed in section 4. Finally, section 5 concludes the paper and presents future works.

2. Background and Related work

In this section, we briefly review the BEB algorithm of the IEEE 802.11 MAC protocol. We also discuss the overheads in transmissions and explain the importance of backoff algorithm. Afterward, several related backoff algorithms are introduced.

2.1. IEEE 802.11 DCF Algorithm

Based on the 802.11 standard, DCF adopts a CSMA/CA mechanism. In this algorithm, each station needs to sense a wireless channel before sending frames. Transmission is performed if the medium is idle for a Distributed InterFrame Space (DIFS) period as shown in Figure 1a – otherwise the backoff procedure is activated. A backoff number is chosen randomly from the interval [0,CW-1]. The backoff number is decremented by one for every idle timeslot during the countdown procedure. The station sends out frames when the backoff number is counted down to zero as shown in Figure 1b. This frame may collide with other frames or be discarded due to wireless error during the transmission. To indicate a successful transmission, an acknowledgement frame (ACK) from the receiver is expected before the ACK timeout timer; otherwise the sender will consider this frame as lost and retransmits it. For each failed transmission, the station doubles its current CW size until CW reaches its maximum value (CW_{max}) (the default value of CW_{max} is 1024 in IEEE 802.11b). The station then performs the backoff procedure again and reduces the probability of collision in the next transmission by using a larger CW (CW_{new}). After a successful transmission, the CW is reset to the initial value (CW_{min}), which is equal to 32 in IEEE 802.11b . The pseudo code for DCF is shown below.



a channel is idle after DIFS

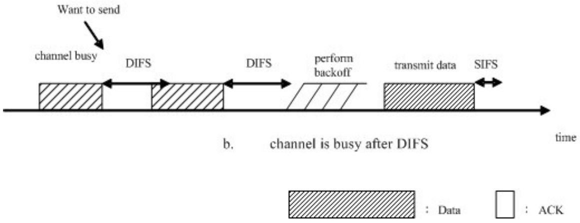


Figure 1. Illustration of the CSMA/CA mechanism

```
if collision
    $CW_{new} = min(CW_{old} * 2, CW_{max})$
if transmission success
    reset CW to $CW_{min}$
```

Pseudo code of DCF

The overheads in this channel access system are composed of idle channel duration, transmission headers, the header overhead and retransmission frames for unsuccessful transmissions. The IFS such as DIFS, Short Inter Frame Space (SIFS) before or after transmitting frames and idle timeslots during the backoff procedure reduce channel utilization. Moreover, transmitting physical and MAC header takes extra time and constrains the system throughput during transmissions [5]. As shown in Figure 2, the length of a physical header is 192 bits in 802.11b if a lone preamble header is applied and the header is transmitted at a rate of 1 Mbps with DPSK modulation. A MAC header (including the CRC checksum) takes 272 bits and can be transmitted at various data rates depending on the applied modulations and channel coding. Furthermore, retransmissions due to collisions or wireless errors make inefficient channel utilization worse within the system. Thus, minimizing the overheads in the backoff procedure directly affects the channel access efficiency.

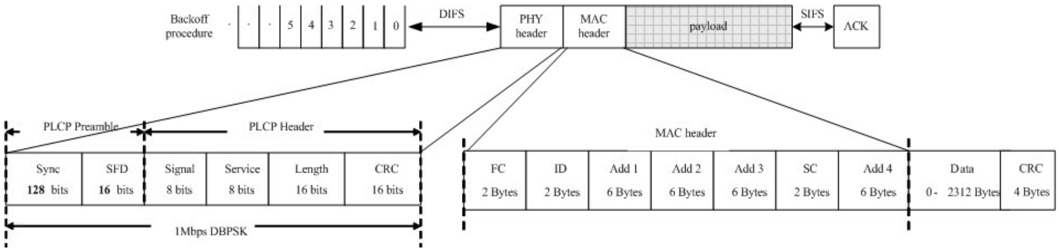


Figure 2. Header format of IEEE 802.11b

2.2 Related Backoff Algorithms

The size of the contention window in the backoff procedure can affect the overheads for network access channels with contentions. A large CW results in a long idle duration when there are only a few active stations in the system (although a large CW could lead to a lower collision rate). A small CW can enhance the channel utilization but the number of collisions could increase quickly if a small CW is used for many active stations. Thus, the backoff algorithm should adapt the correct value of CW to fit the system status. Many past research efforts on enhanced algorithms have been proposed taking into account various considerations such as throughput optimization [7] and QoS requirements [8]. We group these studies into three categories:

- The first category relies on adjusting the CW from each transmission result. The CW is additively/ multiplicatively increased because of collisions and additively/multiplicatively decreased for successful transmissions. The backoff algorithm of DCF, Multiplicative Increase and Linear Decrease (MILD) [9], and EIED [10] algorithms are examples of this category.
- The second category uses a simple estimation method of system status in backoff algorithms. These estimation methods evaluate the network conditions by observing parameters such as latency, jitter, and collision rate. AEDCF [8] is an example of this category.
- The third category applies complex filters in backoff algorithms to estimate the system load. The filter often involves a lot of calculations and assumptions. In [11], the authors measure the channel status, especially occupancy status, and extend the Kalman filter to estimate the number of active stations. We discuss these algorithms in detail below.

a. Exponential Increase Exponential Decrease (EIED)

In an EIED scheme, the station sets the new contention window CW_{new} as CW_{old} multiplied by the parameter r_i, (where r_i is the increment backoff factor) when a collision occurs. As a successful transmission, a new value of CW is given by CW_{old} divided by the parameter r_d where r_d is the decrement backoff factor. The advantage of exponential increase is that Exponential Increase (EI) can reduce the probability of continuous collided frames when many stations contend the channel access concurrently. Moreover, ED (Exponential Decrease) keeps the collision history of the previous transmissions instead of resetting automatically to CW_{min}. ED could prevent numerous collisions from occurring, especially in a network with large number of stations. The pseudocode for EIED is listed below:

```
if  collision
    $CW_{new} = min[CW_{old} * r_i, CW_{max}]$
if  transmission success
    $CW_{new} = max[CW_{old} / r_d, CW_{min}]$
```

Pseudo code of EIED

b. Adaptive Enhanced Coordination Function (AEDCF)

In [8], the authors propose an adaptive service differentiation algorithm for IEEE 802.11 WLANs. Although, the objective of AEDCF is to provide QoS support for multimedia applications by defining different parameter sets for different classes of service, the concept of adjusting CW to network conditions can also be used for enhancing the backoff algorithm of DCF function. In AEDCF each station calculates the collision rate it experiences during a given interval. A high collision rate often indicates the current size of used CW is too small under heavy system loading. Therefore, the station does not reset the CW after a successful transmission. The station sets a new CW based on the current observed collision rate. We take the adapting CW method of highest priority of service classes designed in [8] for explanation. When a collision occurs, the size of CW is doubled to reduce the collision rate. Once a transmission is successful, the station sets the new CW by CW_{old} multiplied by current collision rate (f_{avg}). The pseudocode for AEDCF is listed below:

```
if  collision
    $CW_{new} = min(CW_{old} * 2, CW_{max})$
if  transmission success
    $CW_{new} = max(CW_{old} * MF, CW_{min})$

$MF = min\left(f_{avg}^{j}, 0.8\right)$
$f_{avg}^{j} = (1-\alpha) * f_{curr}^{j} + \alpha * f_{avg}^{j-1}$
```

Pseudo code of AEDCF

In AEDCF, the authors seek to determine an appropriate size of the CW that will minimize collisions and improve the system efficiency. The accuracy of the estimated system status determines whether the CW of the proposed backoff algorithm is appropriate or not. In prior approaches, these algorithms estimate the system load from the number of collisions encountered. This is inadequate because the station only observes its own transmission status which does not represent the whole system condition. The imprecise information affects the system throughput and fairness among stations. In EIED and AEDCF algorithms, a station may inaccurately assign a small CW for a highly contended channel. This will result in this station having a higher channel access opportunity than other stations. When the station with a small CW collides with others, the station will set the CW to a small value once again due to the small CW used in the last transmission and a lower collision rate.

3. Proposed Pause Count Backoff (PCB) algorithm

The main objective of our proposed algorithm is to improve the system performance of DCF with fairness into consideration as well. Previous efforts have focused on improving system efficiency by adjusting the size of CW from transmission results obtained. Although the transmission result correlates with the system status, it cannot precisely determine the system status from a partial transmitting event. Inaccurate estimation results in inefficiency and unfairness among stations. Thus, our proposed PCB algorithm takes global observations to estimate the system status, and then PCB sets an appropriate CW that matches the global system status. There are two steps in our proposed PCB algorithm. The first step is the estimation of CW and the second step involves setting CW.

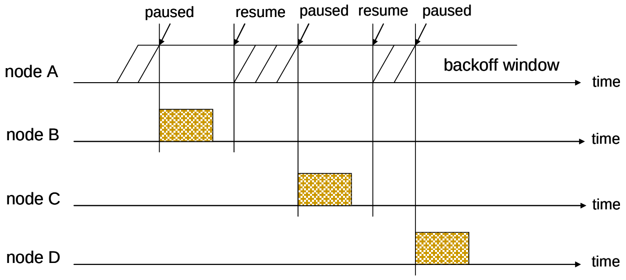


Figure 3. Estimating number of active stations with the paused backoff counter

In step 1 of our proposed PCB approach, a method estimates the number of active stations. In DCF, the backoff counter pauses if other stations transmit frames at the same time and resumes countdown whilst a channel is idle for a DIFS period. The concept is illustrated in Figure 3. The proposed method observes the number of pauses until the counter becomes zero. Each pause represents another station transmitting its frames or more than two stations incurring a collision. The backoff counter is uniformly distributed and the parameter *avg_paused_count* for average number of pauses during the countdown procedure in step 1 could observe the number of active stations in the system. In Figure 3, node A observes three pauses during its backoff procedure. From the observation, node A determines that there are more than three mobile stations concurrently contending for the wireless channel. To keep the scheme stable, Exponential Weighted Moving Average (EWMA) is applied to calculate an average pause count. The number of pauses observed directly correlates to the backoff counter value. A high backoff counter records more pauses than a small backoff counter during the countdown procedure. We propose the solution by setting CW size to active stations in step 2. In step 2, a policy with successful transmission results sets the proper contention window size for the backoff procedure. While transmission is a success, CW is set to the average pause value multiplied by β which relates to the collision rate. The method used to compute β is given below:

$$P_c \cong \frac{2N}{CW} \quad (1)$$

$$\beta = 1 / P_c \quad (2)$$

P_c represents the probability of collision for each transmission in a station. N is the number of active stations in the system. The new CW size can be derived from the number of active stations and β (as shown above). To minimize oscillations in the pause count, the new CW is applied after an observation period which is determined by the number of transmission attempts. In this paper, we set the observation period to 10 transmission attempts in the station. In Figure 4, PCB sets a proper contention window size according to the system status when transmission is successful instead of resetting CW to $CW_{\min}$ immediately or additively/ multiplicatively decreasing CW by a single transmission status. When transmission fails, the new CW should prevent further collisions in the next transmission to reduce the overhead of retransmissions. In our proposed PCB algorithm, the station sets a new CW to a large contention window to avoid further collisions. The station could avoid the next collision effectively by choosing a large backoff counter, although such a large CW may result in more idle timeslots during the backoff procedure. The idea here is that the overheads of collided frames are usually larger than waiting for idle timeslots. The parameter r_d (where r_d is a number for division) used in the pseudocode for the PCB algorithm below determines the size of the new CW. The new CW computed using r_d and $CW_{\max}$ addresses the fairness issue among stations in step 1 when a station does not calculate the system status correctly. The new CW could reset the inappropriate CW and adjust CW again after an observation period.

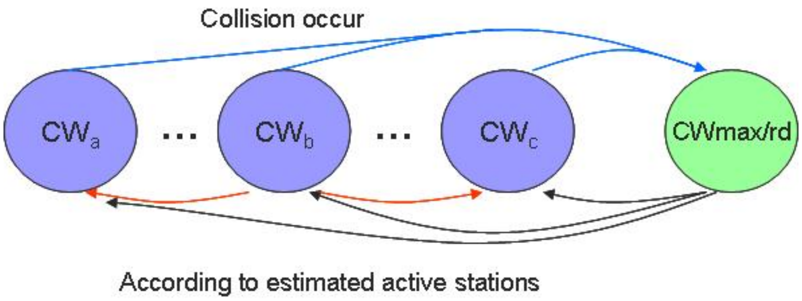


Figure 4. The illustration of contention window adaptation

```
Step 1 : ⟨estimation⟩
    $avg\_pause\_count = (1-\alpha) * avg\_pause\_count + \alpha * current\_pause\_count$

Step 2 : ⟨setting CW⟩
    when collision occurs
        set $CW_{new} = CW_{\max} / r_d$

    when transmission succeeds
        if during observation period
            $CW_{new}$ = current CW
        else
            $CW_{new} = avg\_pause\_count * \beta$
            restart a new observation period
```

Pseudocode of PCB algorithm

4. Performance evaluation

This section evaluates the system performance of the proposed PCB algorithm with other existing backoff algorithms under different system loads. The simulation is performed using NS-2 version 2.28 simulator [12]. Figure 5 shows the network topology used in the simulations. We use IEEE 802.11b based WLAN setup and we assume transmissions without the Request to Send/Clear to Send (RTS/CTS) mechanism in an ideal channel. Each mobile station establishes a Constant Bit Rate (CBR) flow with a 2 Mbps link to the base station. The number of stations is varied from 5 to 50 and the duration of simulation is set to 30 seconds. The parameters used in the simulation are listed in Table 1. The parameters r_i and r_d in EIED algorithm are set to 2 the suggested value in [10]. For AEDCF, we set the observation period of the estimated collision rate to 0.5 seconds and α to 0.8 used in [8]. In the proposed PCB algorithm, we set the weight α to 0.9 to obtain a smooth pause count. The parameter β is set to 5, which means the expected collision rate P_c of a station is around 20% in heavy network load by PCB algorithm. The parameter r_d is set to 4 to get a large size of new CW that is equal to 256 in IEEE 802.11b after collisions.

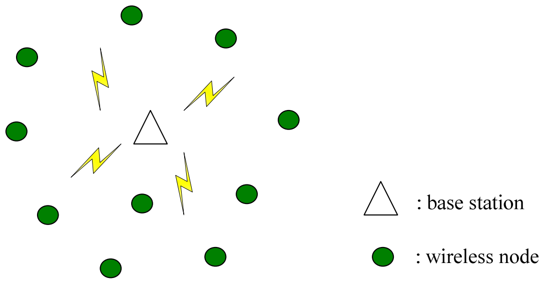


Figure 5. The simulation topology used in our performance evaluation tests

Table 1 IEEE 802.11 b MAC and network parameters used in simulation

Control rate	1 Mbits/s
Data rate	11 Mbits/s
Slot_time	20 μs
SIFS	10 μs
DIFS	50 μs
CW_{min}	32
CW_{max}	1024
Packet size	1000 *bytes*

We use the following performance metrics to evaluate the performance of PCB with other previously proposed algorithms:

- *Goodput*:

Goodput is the most common performance metric that calculates total data amount received in a period by a station. In general, a higher goodput always indicates better efficiency in a system. In this paper, we use aggregate goodput to evaluate PCB and related backoff algorithms.

- *Fairness Index*:

Fairness among stations is an important problem in BEB study and has been discussed by many researchers. Fairness index could show if resource is fairly allocated to each stations. Authors in [13] derived a fairness index using the formula given below:

$$Fairness\ Index = \frac{\left(\sum_i G_i\right)^2}{n * \sum_i \left(G_i\right)^2} \quad (3)$$

Where n is the number of stations and G_i is the goodput of station i achieved. The value of the fairness index is bounded to the interval [0, 1]. The index is equal to 1 when all stations obtain the same goodput.

- *Collision Rate*:

Collision rate gives a probability that packets be discarded due to collisions in each transmission. A higher collision rate usually indicates heavy system load and implies more overheads.

- *Average end-to-end delay*:

End-to-End delay is the time it takes for a packet to travel from sender to receiver. For some time-constraint applications, end-to-end delay is the most concern than other metrics. In this paper, we calculate the average end-to-end delay in the system by various backoff algorithms.

4.1 Goodput Performance

Figure 6 shows the goodput performance results of various backoff algorithms for IEEE 802.11 WLANs. The efficiency of standard DCF performs worse (as expected) when more stations contend for the channel. Although the EIED algorithm takes an exponential decrease CW policy instead of resetting to CW_{min} when there is a successful transmission, the curve decreases when there are more active stations in the system. This means that stations applying EIED and DCF algorithms make decisions with unclear system status and adjust the CW quickly from the result of a single transmission. In contrast to EIED and DCF, the goodput of AEDCF and PCB algorithms remain high with respect to various system loads. These improvements mean that stations using AEDCF and PCB algorithms adjust CW value appropriately according to the load variation within the network. In case of a light system load, when there is a collision, PCB wastes some idle timeslots due to the large new value of CW. The graph of PCB has maximum goodput at around 20 stations. When the number of stations increases, the overheads of collision decreases the efficiency in PCB. Overall, the PCB algorithm obtains a high efficiency compared with other backoff algorithms in various network conditions. Although Figure 6 shows that AEDCF performs consistently better compared to PCB, we demonstrate further below other additional performance (e.g.,fairness, collision rate, average end-to-end delay) benefits provided by PCB over AEDCF.

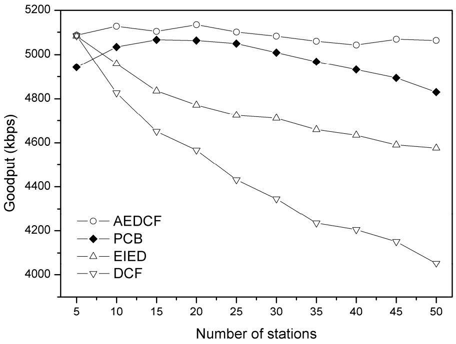


Figure 6. Variation of the aggregate goodput with number of active stations

4.2 Fairness Index

The aggregate goodput in Figure 6 can represent the efficiency of a system. However, when designing a backoff algorithm, we should also consider another important criterion: fairness among stations. The worst case scenario is when one station sets a very small CW due to inaccurate calculations of the system status and as a result always occupies the channel. The aggregate goodput is high but it is unfair to other stations. In Figure 7, we present the fairness index of each backoff algorithm among stations. Using the simulation setup depicted in Figure 5, we executed the simulation for 30 iterations and we calculated a 95% confidence interval. From Figure 7, the proposed PCB algorithm has the most stability and is close a fairness index of one when compared with other contention algorithms. We also observe that the fairness index of AEDCF and EIED are low and oscillatory. This phenomenon means some stations occupy more channel capacity than other stations due to a different understanding of the system status among stations. In the case of the AEDCF algorithm, the estimated collision rate dominates the new CW size. A slight difference in the estimated collision rates among stations at the start of the simulation enlarges the goodput of stations achieved at the end of the experiments.

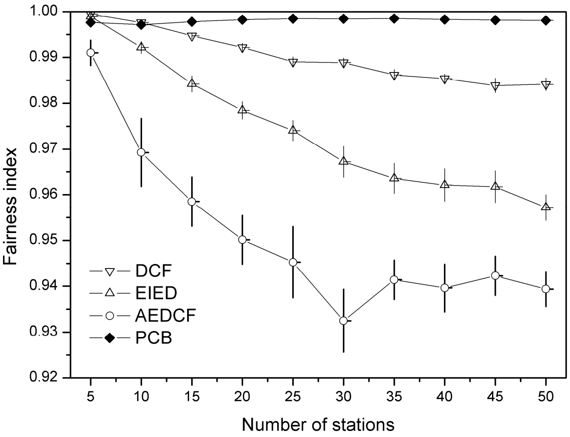


Figure 7. Variation of the fairness index with number of active stations

4.3 Collision Rate

Figure 8 presents the collision rate of various backoff algorithms. As expected, a high collision rate usually indicates additional overheads and longer end-to-end delays. From the simulation results, more frame collisions occurred as the number of active stations in the system grows. The collision rate of a standard DCF is the highest among the compared algorithms due to the resetting behavior for each successful transmission. In the case of n=50, the collision rate is around 50%. The PCB algorithm achieved a low collision rate when compared with other backoff algorithms (as shown in Figure 8). According to the parameter β setting, the collision rate is around 20% as we expected in heavy system load in Figure 8. A low collision rate indicates that the station effectively reduces the retransmission overheads in the system. However, a low collision rate does not mean the least overhead in the system. Even a large CW can result in a low collision rate. In the mean time, the duration of idle channels causes the efficiency of the system to decrease. Therefore, a good backoff algorithm should also examine another metric such as end-to-end delay.

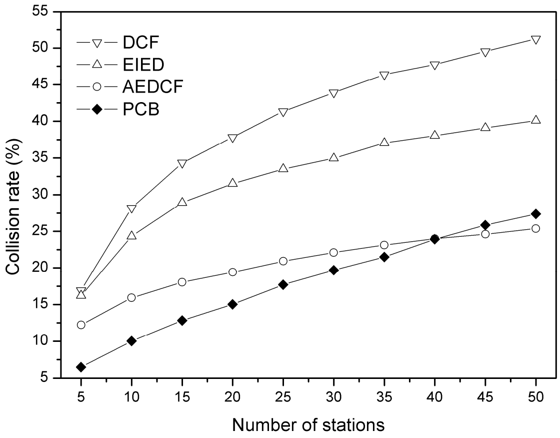


Figure 8. Variation of collision rate with number of active stations

4.4 End-to-End Delay

The variation of the mean end-to-end delay with number of active stations is presented in Figure 9. As expected, the delay increases with the number of stations. The objective of the PCB algorithm is that it precisely estimates the actual network status and sets the corresponding CW to minimize overheads in the system. In Figure 9, the PCB shows the advantage of overhead reduction and obtains the lowest delay among these backoff algorithms. In Figure 10, we present the gain on end-to-end delay by normalizing end-to-end delay of a standard DCF. The delay of the PCB is around 20% less than that of a standard DCF in the case of n = 50. However, the end-to-end delay is too large for multimedia services. The reason for the results is that 802.11b is throughput-oriented without taking QoS requirements into consideration.

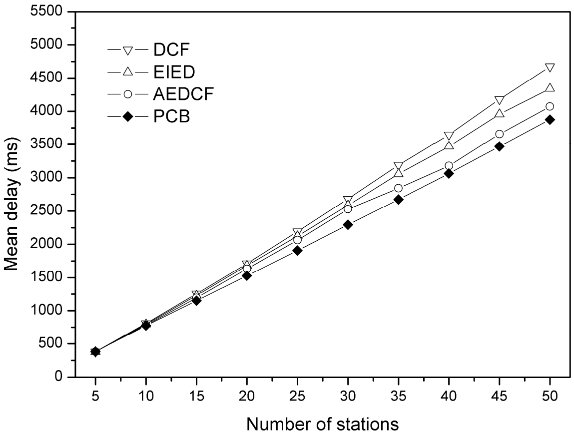


Figure 9. Variation of end-to-end delay with number of active stations

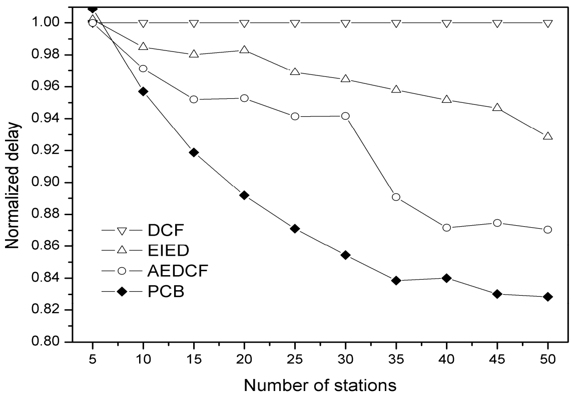


Figure 10. Variation of normalized end-to-end delay with number of active stations

5. Conclusion and future work

In this paper, we proposed a PCB backoff algorithm to improve the efficiency of IEEE 802.11 DCF function. Our PCB algorithm estimates system status by using a pause count backoff counter and determines a proper contention window size that accurately matches current network conditions. We compared the performance of PCB with past proposed algorithms such as IEEE 802.11 DCF, EIED, and AEDCF. Our simulation results demonstrate that PCB outperforms these previously proposed algorithms for various performance metrics and dynamically adapts to the variations in a network. In the future, we plan to consider an extension of this work to consider RTS/CTS mode and wireless transmission errors.

6. References

[1] Information technology - Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks - Speci_c Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Speci_cations. ANSI/IEEE Std 802.11, 1999 Edition.

[2] Andras Veres, Andrew T. Campbell, Michael Barry, and Li_Hsiang Sun, "Supporting Service Differentiation in Wireless Packet Networks Using Distributed Control," IEEE Journal on Selected Areas in Communications, vol. 19, no. 10, October 2001, pp. 2081-2093.

[3] Chonggang Wang, Bo Li, and Lemin Li, "A New Collision Resolution Mechanism to Enhance the Performance of IEEE 802.11 DCF," IEEE Transaction on Vehicular Technology, vol. 53, no. 4, July 2004, pp. 1235-1246.

[4] Shiang-Rung Ye and Yu-Chee Tseng, "A Multichain Backoff Mechanism for 802.11 WLANs," IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, vol. 55, no. 5, September 2006, pp. 1613-1620.

[5] Yang Xiao, "Throughput and Delay Limits of IEEE 802.11," IEEE Communication Letters, vol. 6, no 8, August 2002, pp. 355-357.

[6] Bianchi G., "Performance analysis of the IEEE 802.11 distributed coordination function,"IEEE Journal on Selected Areas in Communications, vol. 18(3), 2000, pp. 535-547.

[7] Lopez Toledo A., Vercauteren T., and Xiaodong Wang;Adaptive, "Optimization of IEEE 802.11 DCF Based on Bayesian Estimation of the Number of Competing Terminals," IEEE Transactions on Mobile Computing, vol. 5, Issue 9, September 2006, pp. 1283 – 1296.

[8] Lamia Romdhani, Qiang Ni, and Thierry Turletti, "Adaptive EDCF: Enhanced Service Differentiation for IEEE 802.11 Wireless Ad-Hoc Networks", in Proc. IEEE WCNC 2003, vol. 2, pp.1373-1378.

[9] V. Bharghavan, "MACAW: A media access protocol for wireless LANs," in Proc. ACM SIGCOMM, 1994, pp. 212-225.

[10] N.-O. Song, B.-J. Kwak, J. Song, and L. E. Miller, "Enhancement of IEEE 802.11 distributed coordination function with exponential increase exponential decrease backoff algorithm," in Proc. IEEE VTC-Sprin, 2003, vol. 4, pp. 2775-2778.

[11] G. Bianchi and I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network," in Proc. IEEE Infocom 2003, Mar. 2003, vol. 2, pp. 884-852.

[12] Network Simulator 2, http://www.isi.edu/nsnam/ns .

[13] R. Jain, D. Chiu, and W. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems," DEC Research Report TR-301, 1984.

Authors

Hao-Ming Liang is currently a Ph.D. candidate studying in the Institute of Computer and Communications, National Cheng Kung University, Tainan, Taiwan. Liang received his M.S. degree from the Institute of Computer and Communications of National Cheng Kung University in 2005. His current research interests include multimedia communications, wireless network, and QoS support.

Sherali Zeadally is an associate professor in the Department of Computer Science and Information Technology at the University of the District of Columbia. He holds BA and MA degrees in Computer Science from University of Cambridge, UK and the PhD degree in Computer Science from University of Buckingham, UK. His research interests include computer networks, mobile computing, RFID, network and system security, and ubiquitous computing. He is a Fellow of the British Computer Society (FBCS) and the Institution of Engineering and Technology (FIET), UK.

Naveen Chilamkurti is currently a lecturer teaching in the Department of Computer Science and Computer Engineering, La Trobe University, Australia. He is a Senior Member of IEEE and has published about 65 conference papers and journal papers. He has a PhD from La Trobe University. His current research areas include wireless Multimedia, Cross-layer techniques, WiMAX, H.264, Wireless QOS, Multicast Security, Reservation Protocols, TCP/IP, Congestion Control, High-Speed Networks, IPv6 Security and Authentication.

Ce-Kuen Shieh is currently a professor teaching in the Department of Electrical Engineering, National Cheng Kung University. He received his Ph.D., MS, and BS degrees from the Electrical Engineering Department of National Cheng Kung University, Tainan, Taiwan. His current research areas include distributed and parallel processing systems, computer networking, and operating systems.