

Deterministic Priority Channel Access Scheme for QoS Support in IEEE 802.11e Wireless LANs

Sunmyeng Kim, Rongsheng Huang, *Student Member, IEEE*, and Yuguang Fang, *Fellow, IEEE*

Abstract—The enhanced distributed channel access (EDCA) of IEEE 802.11e has been standardized to support quality of service (QoS) in wireless local area networks (LANs). The EDCA statistically supports the QoS by differentiating the probability of channel access among different priority traffic and does not provide the deterministically prioritized channel access for high-priority traffic, such as voice or real-time video. Therefore, lower priority traffic still affects the performance of higher priority traffic. In this paper, we propose a simple and effective scheme called deterministic priority channel access (DPCA) to improve the QoS performance of the EDCA mechanism. To provide guaranteed channel access to multimedia applications, the proposed scheme uses a busy tone to limit the transmissions of lower priority traffic when higher priority traffic has packets to send. Performance evaluation is conducted using both numerical analysis and simulation and shows that the proposed scheme significantly outperforms the EDCA in terms of throughput, delay, delay jitter, and packet drop ratio under a wide range of contention level.

Index Terms—Busy tone, enhanced distributed channel access (EDCA), IEEE 802.11e, medium access control (MAC), quality of service (QoS).

I. INTRODUCTION

THE IEEE 802.11 standard for wireless local area networks (WLANs) defines a medium access control (MAC) protocol for sharing the channel among stations [1]. The MAC protocol is designed with two methods of communication for stations: 1) distributed coordination function (DCF) and 2) point coordination function (PCF). The DCF has two data transmission methods: 1) default basic access and 2) optional request-to-send (RTS)/clear-to-send (CTS) access. The basic access method uses the two-way handshaking (DATA-ACK) mechanism. The RTS/CTS access method uses the four-way handshaking (RTS-CTS-DATA-ACK) mechanism to reserve

Manuscript received March 1, 2007; revised August 22, 2007, March 26, 2008, and March 30, 2008. First published May 14, 2008; current version published February 17, 2009. The work of S. Kim was supported in part by the Korean Government (Ministry of Education and Human Resources Development) under Korea Research Foundation Grant KRF-2006-214-D00105. The work of R. Huang was supported in part by the U.S. National Science Foundation under Grant CNS-0721744 and Grant CNS-0626881. The work of Y. Fang was supported in part by the U.S. National Science Foundation under Grant CNS-0721744 and Grant CNS-0626881 and in part by the National Science Council (NSC) under the NSC Visiting Professorship under Contract NSC-96-2811-E-002-010 and Chungwa Telecom under Contract NBY970147. The review of this paper was coordinated by Prof. B. Li.

S. Kim is with the School of Computer and Software Engineering, Kumoh National Institute of Technology, Gumi 730-701, Korea (e-mail: sunmyeng@kumoh.ac.kr).

R. Huang and Y. Fang are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: rshuang@ufl.edu; fang@ece.ufl.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TVT.2008.925319

the channel before transmitting long data packets. This technique is introduced to avoid the hidden terminal problem. The DCF is used to support best-effort data traffic, whereas the PCF supports time-sensitive traffic. The access point (AP) periodically transmits a beacon frame to stations. Between beacon frames, the channel time is divided into a contention-free period (CFP) and a contention period. To eliminate the channel contention among stations, the AP grants channel access to a station by polling the station during the CFP. Stations can only transmit their packets after being polled by the AP.

The widespread use of multimedia applications requires new features, such as high bandwidth and small average delay and delay jitter in WLANs. Unfortunately, the IEEE 802.11 MAC protocol cannot support quality-of-service (QoS) requirements [2]–[6]. The DCF does not differentiate between traffic types [7], and a station might have to wait for an arbitrarily long time to send a packet so that multimedia applications, such as voice and video, may suffer intolerable delay [8].

To support multimedia applications with tight QoS requirements in the IEEE 802.11 MAC protocol, the IEEE 802.11e has been standardized [9]. It introduces a new channel access mechanism called the hybrid coordination function (HCF), which combines functions from the DCF and PCF with some enhancements. The contention-based channel access mechanism of the HCF is referred to as the enhanced distributed channel access (EDCA). The EDCA supports the QoS requirements by introducing four access categories (ACs). Each packet arrives at the MAC layer with priority from the higher layer and is mapped to an AC according to the priority. AC 3, AC 2, AC 1, and AC 0 are for voice, video, best-effort data, and background traffic, respectively. To differentiate the traffic types, the EDCA uses a set of AC specific parameters, which include minimum contention window $CW_{min}[i]$, maximum contention window $CW_{max}[i]$, and arbitration interframe space (AIFS) $AIFS[i]$ for AC i ($i = 0, \dots, 3$). The AIFS is, at least, distributed interframe space (DIFS) long and is calculated with the AIFS number $AIFSN[i]$. The duration of $AIFS[i]$ is defined by $AIFS[i] = SIFS + AIFSN[i] * aSlotTime$, where $SIFS$ is a short interframe space (IFS), and $aSlotTime$ is the duration of a slot time. For $0 \leq i < j \leq 3$, the EDCA has $CW_{min}[i] \geq CW_{min}[j]$, $CW_{max}[i] \geq CW_{max}[j]$, and $AIFSN[i] \geq AIFSN[j]$. Note that, in the preceding inequalities, at least one must be “not equal to.” The EDCA assigns a smaller CW_{min} , CW_{max} , or $AIFS$ to higher priority AC to provide the higher probability to access the channel. Therefore, in the EDCA, the support of QoS can be achieved by differentiating the probability of channel access among different priority ACs [10], [11]. However, there still

remain problems in the EDCA similar to the DCF since it is a contention-based mechanism [12]–[14]. At high loads, there are a large number of collisions, even for high-priority ACs, because low-priority ACs keep attempting to access the channel and collide with high-priority ACs. Therefore, the EDCA does not ensure the QoS requirements [15].

In this paper, we propose a novel distributed contention-based MAC algorithm called deterministic priority channel access (DPCA). The proposed DPCA scheme uses short-duration busy-tone signals (i.e., pulses of energy) [16]–[18] to provide a deterministically prioritized channel access for high-priority ACs and avoid collisions caused by low-priority ACs.

This paper is organized as follows: The related work is presented in Section II. In Section III, the proposed DPCA scheme is presented in detail. The channel time ratio used by voice and data traffic is analyzed in Section IV. In Section V, we discuss the numerical and simulation results. Finally, we draw a conclusion in Section VI.

II. RELATED WORK

To provide QoS guarantee, many studies differentiate traffic types by offering them different QoS parameters or by using busy tones.

Several priority schemes have been studied in the literature over the DCF. Kwon *et al.* proposed QoS support for voice service over IEEE 802.11 WLANs by making a reservation before the data channel access [3], [4]. Xiao proposed a simple priority scheme for real-time applications by differentiating the initial window size, the window-increasing factor, and the maximum backoff stage [2]. Veres proposed a priority scheme that sets different values of minimum and maximum contention windows for different traffic types and different levels of service [19].

The schemes proposed in [2] and [19], as well as the EDCA, statistically support the QoS requirements by differentiating the probability of channel access among different priority traffic types [10], [11], [20]. Although the prioritized channel access is provided in a long-term time scale, it is not guaranteed in a short-term time scale. The reasons are given as follows: A station decreases its backoff counter by one at each slot when the channel is idle after sensing an idle channel for the DIFS/AIFS period. The station transmits its packet, regardless of its priority, when its backoff counter reaches zero. In addition, a high-priority station cannot always be assured to have a smaller backoff counter since the backoff counter is randomly selected based on the uniform distribution. Therefore, lower priority stations can transmit their packets prior to higher priority stations, and the priority of channel access cannot be ensured. As a result, higher priority traffic may wait a long time for the channel contention. The short-term prioritized access is very important for delay- or jitter-sensitive traffic, such as voice and video [21]. Hereinafter, we call this undesirable case the *short-term priority problem*.

Aad and Castelluccia assigned different DIFSs to different priority traffic for service differentiation in the IEEE 802.11 DCF [7]. To ensure that no station with higher priority has packets in its queue when a station with lower priority starts packet transmission, the DIFS for lower priority is set as the

sum of the DIFS and the maximum contention window for higher priority traffic. This scheme can prevent the short-term priority problem previously mentioned. However, when high-priority stations do not have packets to transmit, this scheme results in wasting the channel bandwidth since low-priority traffic stations have to wait for a long time to transmit their packets due to the long DIFS value.

The scheme proposed in [20] makes use of two narrow-band busy-tone signals (i.e., BT1 and BT2). The channel bandwidth is divided into three parts: 1) the BT1 channel; 2) the data channel; and 3) the BT2 channel. Low-priority stations determine the presence of high-priority stations by sensing the carrier on the busy-tone channels. When a high-priority station has a packet to transmit, it will send a BT1 (lasting for one slot time) every M slots during DIFS and backoff stages before it acquires the channel. M is a constant and should be smaller than the IFS of low-priority stations so that they are able to sense the busy-tone signal before they attempt to acquire the data channel. When high-priority stations without packets to transmit and low-priority stations hear this BT1, they will send a BT2 to avoid the hidden terminal problem. All stations with low-priority packets that hear either BT1 or BT2 will defer their transmissions for some duration. This scheme will waste considerable channel bandwidth and energy to send busy-tone signals for each packet transmission. In addition, this scheme just supports two priority traffic types.

In [10] and [11], the authors proposed a deterministic priority access scheme to avoid the short-term priority problem. A station sends a busy tone, instead of its backoff procedure, after sensing an idle channel for the AIFS period. The length of the busy tone is equal to its backoff counter. When the station completes the transmission of the busy tone, it checks the channel status. If the channel is busy, the station defers the current contention. Otherwise, the station transmits its packet. The packet transmission delay of this scheme is larger than that of the EDCA since this scheme, unlike the EDCA, prefers the station with the largest backoff counter, instead of the station with the smallest backoff counter. This scheme also consumes energy to transmit longer busy tones. Furthermore, in the hidden terminal environment, hidden stations cannot sense the other stations' busy tones and send their busy tones so that different priority stations can operate together. Therefore, there exist effects among different priority stations.

In this paper, we propose a DPCA scheme, which uses a busy tone to solve the short-term priority problem. The proposed scheme does not waste the channel bandwidth, even though there are no higher priority traffic. It sends a busy tone twice for each packet transmission so that it can consume much less energy than the schemes in [10], [11], and [20]. In addition, it can have a shorter packet transmission delay compared to the schemes in [10] and [11]. Furthermore, the proposed scheme eliminates effects among different priority traffic types in the hidden terminal environment.

III. DPCA SCHEME

In this section, we describe the priority access scheme that ensures the channel contention based on traffic priority. Our

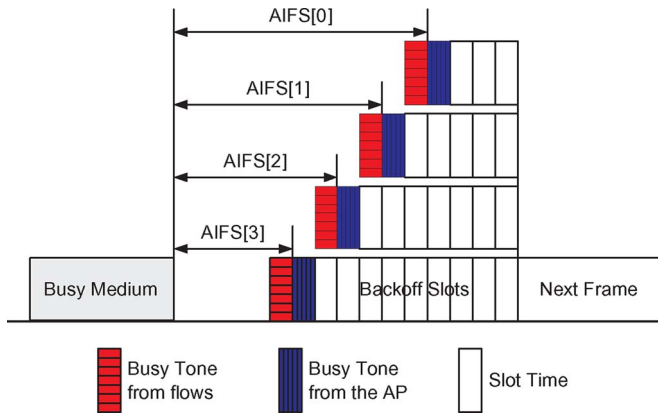


Fig. 1. Busy-tone timing diagram for the DPCA scheme.

scheme needs minor modifications to the EDCA. However, the basic operation of the proposed scheme is the same as that in the EDCA.

To ensure the QoS requirements, high-priority ACs should not be affected by low-priority ACs. To do this, the proposed scheme blocks the transmissions of low-priority ACs by using a busy-tone signal shorter than $aSlotTime$ when high-priority ACs have packets to transmit. In other words, lower priority ACs do not transmit their packets until no higher ACs contend for the channel.

In the EDCA, a flow of a given AC first senses the wireless channel medium. After sensing the idle duration of the AIFS period, the flow waits for random backoff time before transmitting. However, in the proposed scheme, a flow sends a busy tone after sensing an idle channel for the $(AIFS - aSlotTime)$ period (see Fig. 1). On receiving the busy tone from the flow, the AP sends a busy tone at the next time slot so that every flow in a network can recognize the presence of the flow that sends the busy tone. After receiving the busy tone from the AP, the flow operates like the EDCA. Thus, it decreases its backoff counter as long as the channel is sensed idle, does not decrease when a transmission is detected on the channel, and tries to transmit a packet when the backoff counter reaches zero. If the channel is determined to be busy at any time within the $(AIFS - aSlotTime)$ period, then a busy tone and the backoff procedure are suspended. In other words, the flows of lower ACs, which receive a busy tone from either flows of higher AC or the AP, stop their current channel contention and wait until a packet transmission occurs. As long as at least one flow of higher AC exists, all the flows of lower ACs will sense a busy tone within their $(AIFS - aSlotTime)$ periods. The proposed scheme ensures that flows of the highest AC always access the channel.

To let all the flows sense a packet transmission in the hidden terminal environment, the AP sends a negative ACK (NACK) packet for the basic access method or a negative CTS (NCTS) packet for the RTS/CTS access method after receiving a collided packet. Flows that are not in the transmission range of a flow that transmits a packet can detect a packet transmission from the NACK or NCTS packets. This extra signaling cost does not decrease the performance of the proposed scheme since, in the IEEE 802.11e standard, flows wait for the duration

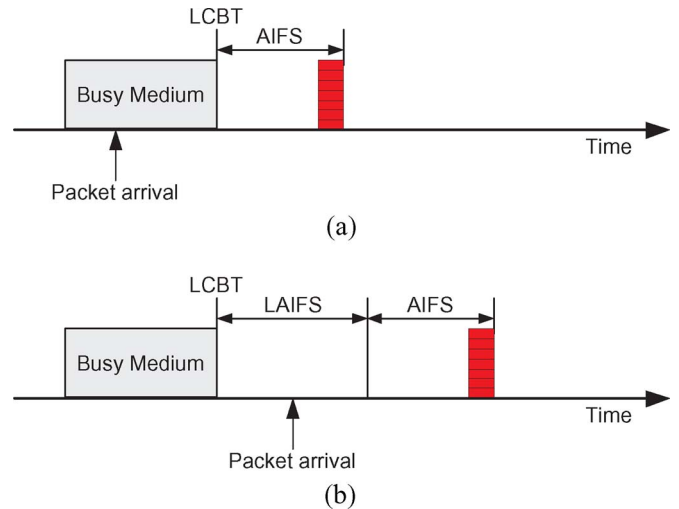


Fig. 2. Busy-tone transmission time according to the packet arrival time. (a) Packet arrival before the LCBT. (b) Packet arrival after the LCBT.

of the $(EIFS - DIFS + AIFS)$ period to start the backoff operation when a packet collision occurs, and the $(EIFS - DIFS)$ value is equal to the time needed to transmit a NACK or an NCTS packet, where the EIFS is an extended IFS.

For the operation of the proposed scheme, let us assume that, unlike the EDCA, a higher priority AC always has a smaller AIFSN than a lower priority AC, i.e., $AIFSN[i] > AIFSN[i + 1] + 1$ for $0 \leq i \leq 2$.

The busy-tone transmission time varies with the packet arrival time and channel status. To determine the time, the proposed scheme uses three parameters: 1) the packet arrival time (PAT) at the MAC layer; 2) the last channel busy time (LCBT) due to the recent packet transmission; and 3) the AIFS of the lowest priority AC (LAIFS). The LCBT is set to the completion time of a packet transmission (i.e., the end time of an ACK, a NACK, or an NCTS packet).

There are two possible cases that determine the transmission time of a busy tone (see Fig. 2). If a flow receives a busy tone between LCBT and PAT, it defers its operation until it senses the channel busy by a packet transmission. Otherwise, it operates as in the following two cases: First, in the case where a flow receives a new packet from the upper layer before the LCBT, it sends its busy tone after sensing an idle channel for the $(AIFS - aSlotTime)$ period [see Fig. 2(a)]. Second, when a packet arrives after the LCBT, the channel should be determined to be idle until $(LCBT + LAIFS * N + AIFS - aSlotTime)$ before the busy tone is allowed to transmit [see Fig. 2(b)], where N is used to align the start time of the AIFS period with the integral multiples of LAIFS and is $\lceil (PAT - LCBT) / LAIFS \rceil$. $\lceil x \rceil$ rounds to the smallest integer greater than or equal to x . This alignment is needed to ensure the channel contention to be among flows with the same priority AC. If a flow sends its busy tone after sensing the idle duration of the $(AIFS - aSlotTime)$ period without the alignment, it may cause a busy-tone collision with flows of other priority ACs. Then, the flows with the collided busy tone contend for the channel at the same time so that there exist effects among different priority ACs. Before the busy-tone transmission time expires, if a busy tone is detected from other flows or the AP,

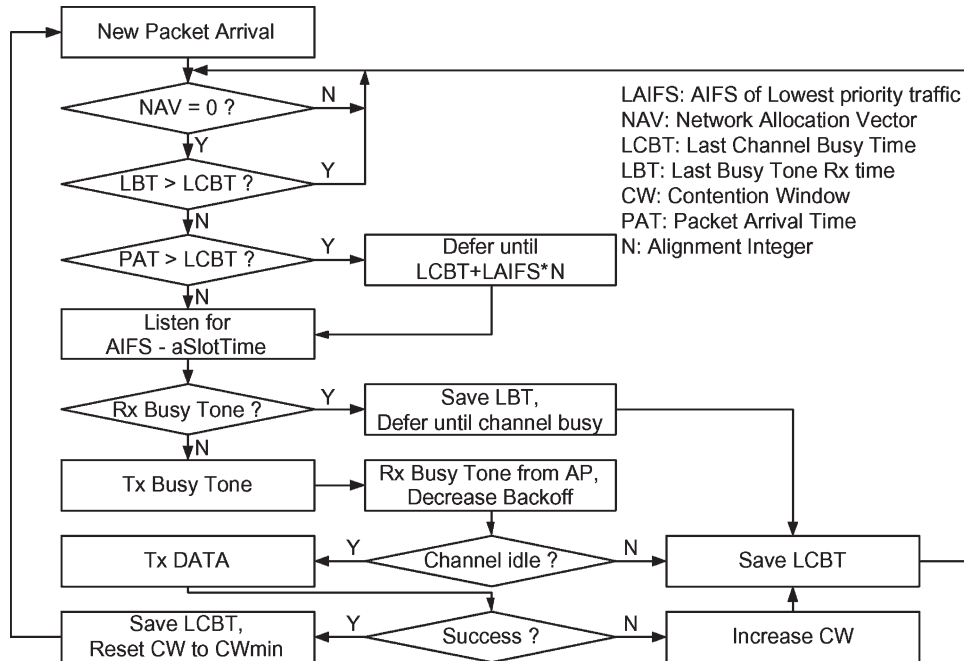


Fig. 3. Operation procedure of the DPCA scheme.

then a flow should defer its operation until a packet transmission occurs and operate as the first case. Fig. 3 shows the operation procedure of the proposed scheme.

In the DPCA scheme, distinguishing a busy tone from a packet transmission is very important to guarantee the proper operation. To do this, the duration of a transmission is used. The transmission time for a packet has a duration of at least three time slots, because it includes the physical preamble and header of $20 \mu s$, which is from Table I in Section V. A busy-tone duration is smaller than one time slot. Estimating the duration is simple without any additional overhead or cost, because every station performs carrier sensing. Each station, by using carrier sensing, observes the channel status and measures the duration of the busy period. Therefore, the proposed scheme can discriminate between a busy tone and a packet transmission when receiving a signal.

IV. PERFORMANCE ANALYSIS

In the proposed scheme, each priority traffic separately performs its own packet transmission operation and is provided the deterministically prioritized channel access. As loads of higher priority traffic types increase, they use the larger amount of channel time to meet their QoS requirements and affect the performance of lower priority traffic types. In this section, we analyze the channel time usage ratio, which is the fraction of time during which the channel is used to transmit each priority traffic. We assume that all the stations are in the transmission range of one another so that whenever a station is transmitting, all the other stations can detect this transmission, i.e., there is no hidden terminal problem.

We denote a *transmission interval* as the time duration between two consecutive packet transmissions, which is made up of four components: 1) AIFS; 2) busy tone from the AP; 3) backoff; and 4) transmission (see Fig. 4).

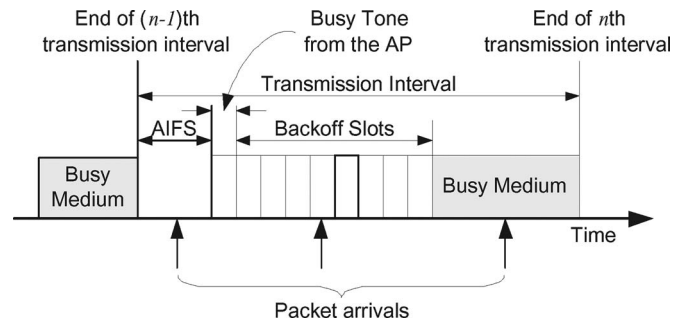


Fig. 4. Structure of the transmission interval.

We consider a WLAN supporting two types of traffic: 1) voice and 2) data. Voice traffic is assigned a higher priority. Suppose there are fixed N_v voice flows and N_d data flows admitted in the WLAN. For data traffic, each flow always has packets to transmit, and each packet has to wait for random backoff time before being transmitted. Data flows share the channel time unused by voice flows.

Under the considered environment, a flow, which has a packet arrival during the ongoing transmission interval, does not contend for the channel. For example, in Fig. 4, there are three new packet arrivals of different flows. The first packet arrives during the AIFS period and has to wait until $(LCBT + LAIFS + AIFS - aSlotTime)$ [see Fig. 2(b)]. It will receive a busy tone before the time expires since there is always at least one flow with a packet to transmit. For the second packet, it already received a busy tone from other flows before arriving. For the third packet, it defers its access operation since the channel is sensed busy. Therefore, only flows with a packet arrival before the LCBT contend for the channel.

The proposed scheme limits the operation of flows of lower priority ACs by busy tones. Therefore, only flows with the same priority AC contend for the channel during a transmission interval.

To calculate the transmission interval, the AIFS, busy tone time, and transmission time can easily be obtained, but backoff duration varies with time, because it depends on how many flows are contending for the channel during the current ongoing transmission interval. The number of data flows is always N_d because of the saturation condition. The number of voice flows varies with time under the nonsaturated condition. The number of voice flows, which will contend for the channel during the next transmission interval, is determined at the end of a packet transmission, regardless of a successful transmission or a collision. The number of contending voice flows is described by a process with state $\{0, 1, 2, \dots, N_v\}$. After each transmission interval, a state will remain in the current state or move to the next state. For example, in Fig. 4, assume that there are n_v voice flows with a packet at the end of the $(n - 1)$ th transmission interval, and three new packets of different voice flows arrive during the n th interval. At the end of the n th interval, the number of contending flows becomes $n_v + 2$ (if the n th transmission is successful) or $n_v + 3$ (if collided).

We can express the channel time usage ratio in terms of stationary probability S_i and the corresponding duration of transmission interval D_i at state i ($0 \leq i \leq N_v$). As no voice flows contend for the channel, data flows will occupy the channel. D_0 is the duration of the transmission interval for data flows. The ratios for voice U_v and data U_d are clearly given by

$$U_v = \frac{\sum_{i=1}^{N_v} S_i \cdot D_i}{\sum_{i=0}^{N_v} S_i \cdot D_i}. \quad (1)$$

$$U_d = 1 - U_v. \quad (2)$$

Our analysis is divided into two parts: First, we study the behavior of the state model and obtain the stationary probability and the corresponding duration of transmission interval for voice traffic. Then, we find the duration for data traffic.

A. Stationary Probability and Duration of Transmission Interval for Voice Traffic

We make several assumptions. First, voice packets do not accumulate in the transmission queue, which means that each voice flow has up to one voice packet. Therefore, hereinafter, a packet arrival means that a voice flow can start contending for the channel. Second, we do not consider the retry limit. Third, the maximum contention window is equal to the minimum contention window for voice traffic. We define, for convenience, $CW = CW_{min} = CW_{max}$. These assumptions are reasonable, because of the following: Voice traffic does not collide with data traffic so that its collision probability remains low in steady state. Therefore, a packet can be transmitted within the packet interarrival time, and a voice flow can maintain the packet drop probability at a relatively small value. The IEEE 802.11e EDCA standard recommends that the values of CW_{min} and CW_{max} for voice traffic are set to $(aCW_{min} + 1)/4 - 1$ and $(aCW_{min} + 1)/2 - 1$, respectively, where aCW_{min} is the minimum contention window for data traffic [9]. Therefore, the difference between CW_{min} and CW_{max} is so small that its effect is negligible.

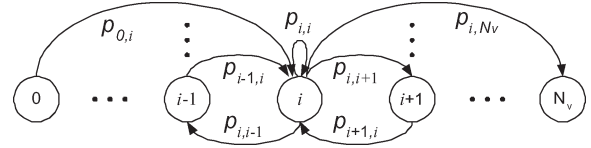


Fig. 5. State transition diagram for voice traffic.

In Fig. 5, we use the state transition diagram to model the number of contending voice flows. Here, we explain the transitions only for state i . Any number of voice packets less than or equal to $N_v - i$ can arrive, but only one packet can successfully be transmitted during a transmission interval. Therefore, state i can enter any state above it and but can only enter state $i - 1$ for states lower than i . Similarly, state i is entered from any state below it and from state $i + 1$. State i can also remain in the same state.

Packets from a voice flow arrive at constant interval T . The start time for each voice flow is randomly given. Therefore, we assume that the arrival probability of a voice packet is D_i/T at state i , where D_i is less than T . We also assume that packets collide with one another with a constant and independent probability $P_{c,i}^v$ at state i , regardless of the number of retransmissions, and succeed with probability $P_{s,i}^v$.

The state transition diagram has seven transition probabilities.

- 1) There are no new packet arrivals at state 0, i.e.,

$$p_{i,i} = \left(1 - \frac{D_i}{T}\right)^{N_v}, \quad i = 0.$$

- 2) There is a successful transmission and one new packet arrival, or there is a collision and no new packet arrivals, i.e.,

$$p_{i,i} = P_{s,i}^v \cdot \binom{N_v - i}{1} \left(\frac{D_i}{T}\right) \left(1 - \frac{D_i}{T}\right)^{N_v - i - 1} + P_{c,i}^v \cdot \left(1 - \frac{D_i}{T}\right)^{N_v - i}, \quad 1 \leq i \leq N_v - 1.$$

- 3) There is an unsuccessful transmission at state N_v , i.e.,

$$p_{i,i} = P_{c,i}^v, \quad i = N_v.$$

- 4) There is a successful transmission and no new packet arrivals, i.e.,

$$p_{i,i-1} = P_{s,i}^v \cdot \left(1 - \frac{D_i}{T}\right)^{N_v - i}, \quad 1 \leq i \leq N_v.$$

- 5) There are one or more new packet arrivals at state 0, i.e.,

$$p_{i,j} = \binom{N_v}{j} \left(\frac{D_i}{T}\right)^j \left(1 - \frac{D_i}{T}\right)^{N_v - j}, \quad i = 0, 1 \leq j \leq N_v.$$

- 6) There are $(j - i + 1)$ new packet arrivals when a transmission is successful or $(j - i)$ new packet arrivals when collided, i.e.,

$$p_{i,j} = P_{s,i}^v \cdot \binom{N_v - i}{j - i + 1} \left(\frac{D_i}{T}\right)^{j-i+1} \left(1 - \frac{D_i}{T}\right)^{N_v - j - 1} \\ + P_{c,i}^v \cdot \binom{N_v - i}{j - i} \left(\frac{D_i}{T}\right)^{j-i} \left(1 - \frac{D_i}{T}\right)^{N_v - j} \\ 1 \leq i \leq N_v - 2 \quad i + 1 \leq j \leq N_v - 1.$$

- 7) There are $(j - i)$ new packet arrivals when a transmission is collided, i.e.,

$$p_{i,j} = P_{c,i}^v \cdot \left(\frac{D_i}{T}\right)^{j-i}, \quad 1 \leq i \leq N_v - 1 \quad j = N_v.$$

After each transmission, state i transits to state j ($j = i$: transition probabilities 1–3, $j = i - 1$: transition probability 4, $j > i$: transition probabilities 5–7).

We can derive the stationary probability S_i that the state is in i , which is a recursive expression and can be given in terms of a single unknown constant S_0 , i.e.,

$$S_i = \frac{1 - p_{i-1,i-1}}{p_{i,i-1}} S_{i-1} - \frac{\sum_{j=0}^{i-2} p_{j,i-1} S_j}{p_{i,i-1}}, \quad 0 < i \leq N_v. \quad (3)$$

S_0 is determined by the normalization equation, i.e.,

$$\sum_{i=0}^{N_v} S_i = 1. \quad (4)$$

Now we calculate the average number of backoff slots at state i . Let $p(i, l, k)$ be the probability that l voice flows choose backoff counter k and the other contending flows choose the counter larger than k at state i . Then, we have

$$p(i, l, k) = \binom{i}{l} \left(\frac{1}{CW + 1}\right)^l \left(\frac{CW - k}{CW + 1}\right)^{i-l}. \quad (5)$$

From (5), we obtain the probability $p(i, k)$ that, at state i , the number of backoff slots is k when any transmission occurs, i.e.,

$$p(i, k) = \sum_{l=1}^i p(i, l, k). \quad (6)$$

The average number of backoff slots $T_{bo,i}^v$ is calculated from (6) as follows:

$$T_{bo,i}^v = \sum_{k=0}^{CW} p(i, k) \cdot k. \quad (7)$$

The average duration of transmission interval D_i is given by

$$D_i = AIFS[voice] + T_{bo,i}^v \cdot \sigma + P_{s,i}^v \cdot T_s^v + P_{c,i}^v \cdot T_c^v + BT \quad (8)$$

where $P_{s,i}^v$ is the probability of a successful transmission that exactly one flow transmits and that the remaining flows defer transmissions, and $P_{c,i}^v$ is the collision probability. T_c^v and T_s^v are the average time intervals that the channel is sensed busy due to collision and successful transmission, respectively. σ is the duration of a time slot. BT is the time duration for a busy tone from the AP (i.e., $= \sigma$). $P_{s,i}^v$ and $P_{c,i}^v$ are given by

$$P_{s,i}^v = \sum_{k=0}^{CW} p(i, 1, k) \quad (9)$$

$$P_{c,i}^v = 1 - P_{s,i}^v \quad (10)$$

respectively.

For the basic and RTS/CTS access methods, T_c^v and T_s^v are given by

$$\begin{cases} T_{c,basic}^v = H + L + SIFS + NACK + 2\delta \\ T_{s,basic}^v = H + L + SIFS + ACK + 2\delta \end{cases} \quad (11)$$

$$\begin{cases} T_{c,rts/cts}^v = RTS + SIFS + NCTS + 2\delta \\ T_{s,rts/cts}^v = RTS + SIFS + CTS + SIFS \\ \quad + H + L + SIFS + ACK + 4\delta \end{cases} \quad (12)$$

where $H(= PHYhdr + MACHdr)$ is the time to transmit a packet header; δ is the propagation delay; $SIFS$ is the SIFS time space; RTS , CTS , ACK , $NACK$, and $NCTS$ are the times to transmit an RTS, CTS, ACK, NACK, and NCTS, respectively; and L is the packet transmission time.

B. Duration of Transmission Interval for Data Traffic

In the proposed scheme, data flows contend for the channel when there are no contending voice flows. During the ongoing transmission interval, new arriving voice packets have to suspend their busy-tone transmission and backoff procedure until the completion of the current successful transmission or collision of data flows. Therefore, D_0 is the duration of transmission interval only for data traffic.

To calculate the duration, we use the same Markov chain model proposed in [22] and [23]. The analysis of the proposed scheme can be obtained from the analysis in [22] and [23]. Therefore, we omit the derivations of the equations that can straightforwardly be derived along the lines of [22] and [23].

P_{tr} is the probability that at least one flow transmits a packet in a given slot time, and P_s^d is the probability of a successful transmission that exactly one flow transmits and that the remaining $N_d - 1$ flows defer transmissions. P_c^d is the collision probability. T_c^d and T_s^d are the average times that the channel is sensed busy due to collision and successful transmission, respectively. τ is the transmission probability that a flow transmits a packet in a randomly chosen slot time [22], [23].

TABLE I
NUMERICAL AND SIMULATION PARAMETERS

Parameter	Value
Data Bit Rate	54 Mbps
Control Bit Rate	6 Mbps
Slot Time	9 μ s
SIFS	16 μ s
Retry Limit	7
Propagation Delay	1 μ s
MAC Header	26 Octets
FCS	4 Octets
PHY PLCP Preamble Length	16 μ s
PHY PLCP Header Length	5 Octets
ACK	14 Octets

TABLE II
TRAFFIC PARAMETERS

Parameter	Voice	Video	Data
$AIFS_N$	2	4	7
CW_{min}	7	15	31
CW_{max}	15	31	1023
Packet Size (Octets)	80	1000	1500
Inter Arrival Time (ms)	10	12.5	12.5
Sending Rate (Kbps)	64	640	960

P_{tr} , P_s^d , and P_c^d are given by

$$P_{tr} = 1 - (1 - \tau)^{N_d} \quad (13)$$

$$P_s^d = \frac{N_d \tau (1 - \tau)^{N_d - 1}}{P_{tr}} = \frac{N_d \tau (1 - \tau)^{N_d - 1}}{1 - (1 - \tau)^{N_d}} \quad (14)$$

$$P_c^d = 1 - P_s^d \quad (15)$$

respectively.

The average number of backoff slots per transmission T_{bo}^d is given as follows:

$$T_{bo}^d = \frac{1 - P_{tr}}{P_{tr}}. \quad (16)$$

The average duration of the transmission interval for data traffic D_0 is given by

$$D_0 = AIFS[data] + T_{bo}^d \cdot \sigma + P_s^d \cdot T_s^d + P_c^d \cdot T_c^d + BT \quad (17)$$

where T_c^d and T_s^d can be obtained from (11) and (12).

Substituting (3), (8), and (17) into (1), we can obtain the channel time usage ratio.

V. NUMERICAL AND SIMULATION RESULTS

In this section, we discuss the numerical and simulation results of the proposed DPCA scheme. To study the performance of the DPCA scheme and validate the accuracy of the analytical model, we have implemented it with the NS-2 simulator. We compare them to the results of the IEEE 802.11e EDCA and guaranteed priority and enhanced fairness (GPEF), which is proposed in [10] and [11]. System parameters used in the numerical analysis and simulation are listed in Table I.

We simulated an IEEE 802.11a network with transmission rates of 54 Mb/s for data packets and 6 Mb/s for control packets such as ACK, respectively. We have three types of traffic: 1) voice; 2) video; and 3) data. The traffic parameters are listed in Table II. A constant bit rate model is used for all three traffic types. For voice traffic, the overhead (40 octets), such as the

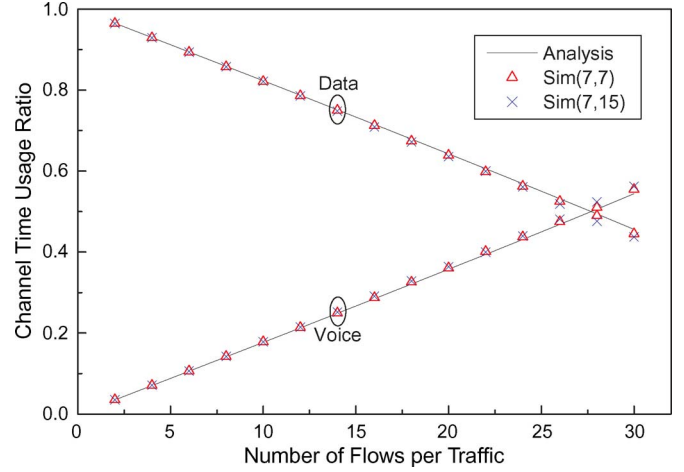


Fig. 6. Numerical and simulation results for the channel time usage ratio.

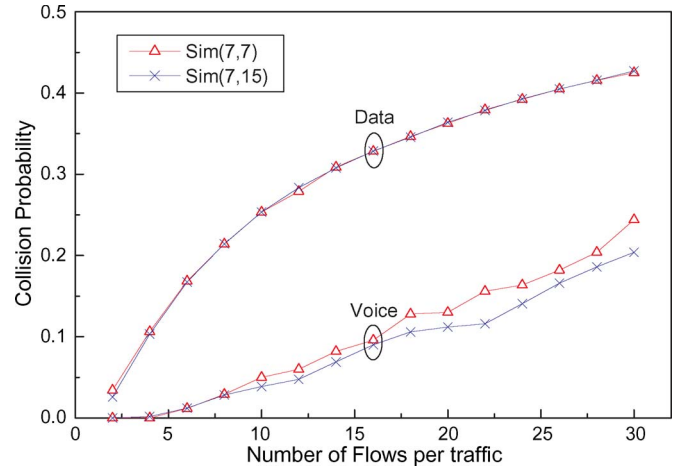


Fig. 7. Collision probability when ($CW_{min} = 7, CW_{max} = 7$) and ($CW_{min} = 7, CW_{max} = 15$) for voice traffic.

RTP/UDP/IP headers, is added so that the sending rate becomes 96 kb/s at the MAC layer. In the simulation, we consider the basic access method and only uplink traffic. In addition, we assume that each station has a single flow of voice, video, or data traffic. Simulations run for 100 s, and all simulation results are averaged over ten simulations.

Main performance metrics of interest are throughput, collision probability, average delay, delay jitter, drop probability, and channel time usage ratio. Delay is the time elapsed from the moment a packet arrives at the MAC layer queue until the packet is successfully transmitted to the intended station, including the receipt of acknowledgement, and delay jitter is the standard deviation of the delay. Drop probability is the ratio between the number of packets dropped due to the retry limit and the total number of data packets being exceeded.

A. Single-Hop Topology

In this scenario, we consider one WLAN where all stations are in the transmission range of one another. In the simulation, all three traffic types have the same number of flows.

Fig. 6 shows the numerical and simulation results for channel time usage ratios. Fig. 7 shows the collision probability

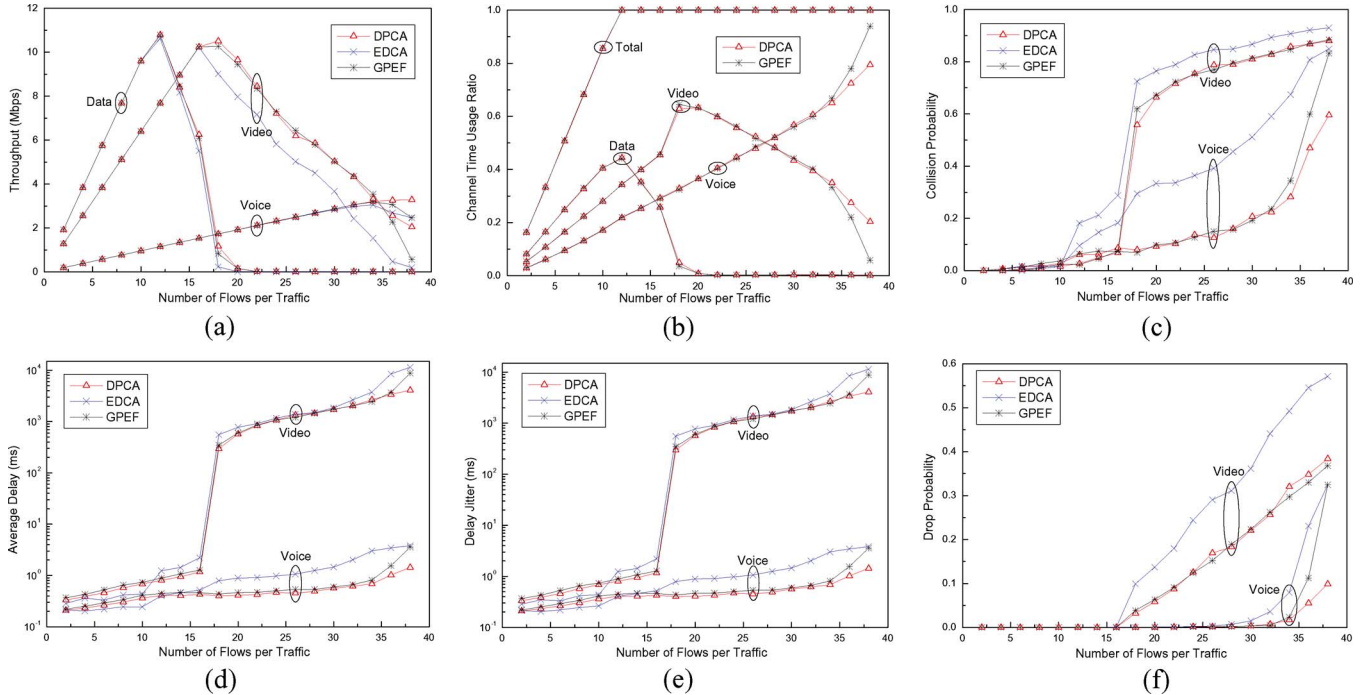


Fig. 8. Performance according to the number of flows per traffic in a single hop. (a) Throughput. (b) Channel time usage ratio. (c) Collision probability. (d) Average delay. (e) Delay jitter. (f) Packet drop probability.

according to the minimum and maximum contention windows for voice traffic. For voice and data, we use the same values as traffic parameters in Table II. However, we change the interarrival time of data traffic to a small value for the saturation condition. In these figures, Sim(7,7) and Sim(7,15) mean the simulation results with ($CW_{min} = 7, CW_{max} = 7$) and ($CW_{min} = 7, CW_{max} = 15$) for voice traffic, respectively. Fig. 6 shows a close match between the numerical and simulation results. There is also no difference between Sim(7,7) and Sim(7,15). Sim(7,15) makes collision probability slightly lower than Sim(7,7), as shown in Fig. 7. However, this may also make the average backoff time longer. Therefore, the channel time usage ratio is almost similar. From Fig. 6, we also observe that voice traffic is only allowed to occupy a certain channel time, whereas the remaining channel time is dedicated to data traffic.

Fig. 8(a) shows the effect of the number of flows on the throughput, where for data traffic, we see that there is no performance difference among the DPCA, GPEF, and EDCA schemes at light loads. However, as the number of flows increases, the difference becomes noticeable such that, in the EDCA, the throughput for data sharply goes to zero, whereas, in the DPCA and GPEF, it slowly decreases compared with the EDCA. As for video traffic, almost the same behaviors as data traffic can be observed. When the load becomes higher, the throughput for the EDCA becomes worse than the other schemes, particularly over the range where the throughput for data traffic becomes zero. The performance difference between the DPCA and GPEF becomes markedly noticeable at high loads. For voice traffic, the throughput of the EDCA soon becomes saturated and decreases as the number of flows becomes larger. This is because of the fact that, in the EDCA, voice cannot gain exclusive access over video and data, and data and video traffic types still try to access the channel and collide with voice traffic. At high loads, the

throughput of the GPEF becomes rapidly worse. However, the throughput of the DPCA becomes saturated at the point with a larger number of flows.

Fig. 8(b) shows the channel time usage ratio for the DPCA and GPEF schemes. Data and video traffic types have almost the same behaviors, as shown in Fig. 8(a). The reason for this is that, to guarantee the QoS requirements for higher priority traffic types, the DPCA and GPEF schemes discriminate lower traffic and reallocate the amount of saved channel time to higher priority traffic. From this figure, we can also observe that the voice flows obtain a significant portion of the channel time as the number of voice flows increases. At high loads, for voice traffic, the GPEF needs more channel time than the DPCA since it prefers the station with the largest backoff counter. Therefore, the channel time for the video traffic of the GPEF dramatically decreases.

Fig. 8(c) shows the results for collision probability. Hereinafter, we omit the results for data traffic, because it requires high throughput, but other QoS metrics are less stringent than voice or video traffic. In the EDCA, collision probability gets higher as the number of flows becomes larger since all priority traffic types always try to access the channel and cause collisions with one another. In the DPCA and GPEF, collision probability slowly increases compared with the EDCA. At high loads, most voice flows contend one another to access the channel. Therefore, the collision probability of the DPCA and GPEF increases like the EDCA. However, the GPEF is more steep than the DPCA. The reason is given as follows: The GPEF prefers the station with the largest backoff counter so that more voice packets are queued. Therefore, the channel contention probability increases.

Fig. 8(d) and (e) shows the results for average delay and jitter, respectively, where we can see that, at light loads, the

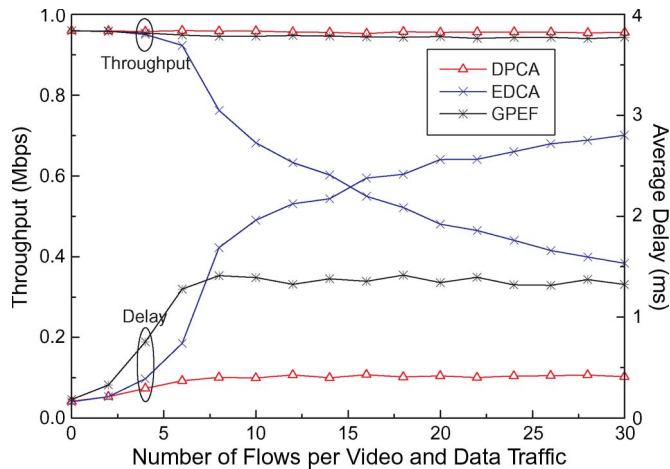


Fig. 9. Throughput and average delay of voice traffic in the hidden terminal environment.

delay and jitter for the DPCA and GPEF schemes are slightly worse than those for the EDCA. There are two reasons for the DPCA: First, if a packet arrives after the LCBT, the flow has to wait for a longer time than its AIFS to send a busy tone, as shown in Fig. 2(b). Second, when a flow receives a busy tone from other priority traffic, it defers its backoff operation until sensing a packet transmission, although the channel is already sensed idle. For the GPEF, it is because this scheme prefers the flow with the largest backoff counter. However, the delay and jitter differences between three schemes are small, and the QoS requirements can be met at light loads. As the load increases, the DPCA outperforms the EDCA and GPEF for all priority traffic types, which is another advantage of our scheme.

Fig. 8(f) shows the results for drop probability. It can be observed that there are fewer packet drops in the DPCA. On the contrary, the EDCA and GPEF cause many packet drops due to its high collision probability, as shown in Fig. 8(c), particularly when there are many flows in the network.

B. Hidden Terminal Topology

To validate the proposed scheme in the hidden terminal environment, we simulated a network where the flows of each AC are divided into two groups that are not in the transmission range of one another. In the simulation, voice traffic has ten flows, and video and data traffic types have the same number of flows.

Fig. 9 shows the throughput and average delay of voice traffic according to the number of video and data flows. For the EDCA, the throughput and average delay become worse as the number of video and data flows increases since all priority flows always try to access the channel and make collisions with one another. For the DPCA and GPEF, the throughput remains the same and always met with voice traffic's requirement, regardless of the number of video and data flows. The DPCA always has better throughput performance than the GPEF. The average delay of the DPCA is kept low and stable. However, it slightly increases when there exist video and data traffic types. This is from the fact that, in the proposed scheme, voice flows with a new packet arrival during transmission intervals of video or data flows must wait until sensing a packet transmission.

VI. CONCLUSION

IEEE 802.11e EDCA provides only statistically prioritized channel access. Therefore, the EDCA does not completely ensure the QoS requirements in practice. In this paper, we proposed the DPCA to improve the QoS performance over the EDCA. The proposed DPCA scheme limits the operations of lower priority traffic through a busy tone. A deterministically prioritized channel access is provided to higher priority traffic. We analyze the channel time usage ratio under voice and data traffic types. The simulation results show that the DPCA is very effective and has significantly higher throughput and lower delay and delay jitter by keeping the collision probability low, regardless of the variation of contention levels. Moreover, the results indicate that our scheme can provide the required QoS performance to multimedia applications.

REFERENCES

- [1] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std. 802.11, Jun. 1999.
- [2] Y. Xiao, "A simple and effective priority scheme for IEEE 802.11," *IEEE Commun. Lett.*, vol. 7, no. 2, pp. 70–72, Feb. 2003.
- [3] Y. Kwon, Y. Fang, and H. Latchman, "Design of MAC protocols with fast collision resolution for wireless local area networks," *IEEE Trans. Wireless Commun.*, vol. 3, no. 3, pp. 793–807, May 2004.
- [4] Y. Kwon, Y. Fang, H. Latchman, "A novel MAC protocol with fast collision resolution for wireless LANs," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, San Francisco, CA, Mar./Apr. 2003, pp. 853–862.
- [5] H. Zhai, X. Chen, and Y. Fang, "How well can the IEEE 802.11 wireless LAN support quality of service?" *IEEE Trans. Wireless Commun.*, vol. 4, no. 6, pp. 3084–3094, Nov. 2005.
- [6] H. Zhai, Y. Kwon, and Y. Fang, "Performance analysis of IEEE 802.11 MAC protocols in wireless LANs," *Wireless Commun. Mobile Comput.*, vol. 4, no. 8, pp. 917–931, Dec. 2004.
- [7] I. Aad and C. Castelluccia, "Differentiation mechanisms for IEEE 802.11," in *Proc. IEEE INFOCOM*, Apr. 2001, vol. 1, pp. 209–218.
- [8] Y. Xiao, H. Li, and S. Choi, "Protection and guarantee for voice and video traffic in IEEE 802.11e wireless LANs," in *Proc. IEEE INFOCOM*, Mar. 2004, vol. 3, pp. 2152–2162.
- [9] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment: Medium Access Control (MAC) Quality of Service Enhancements*, IEEE Std. 802.11e, Nov. 2005.
- [10] H. Jiang, P. Wang, and W. Zhuang, "A distributed channel access scheme with guaranteed priority and enhanced fairness," *IEEE Trans. Wireless Commun.*, vol. 6, no. 6, pp. 2114–2125, Jun. 2007.
- [11] P. Wang, H. Jiang, and W. Zhuang, "IEEE 802.11e enhancement for voice service," *IEEE Wireless Commun.*, vol. 13, no. 1, pp. 30–35, Feb. 2006.
- [12] A. Grilo and M. Nunes, "Performance evaluation of IEEE 802.11e," in *Proc. IEEE PIMRC*, Sep. 2002, vol. 1, pp. 511–517.
- [13] L. Romdhani, Q. Ni, and T. Turletti, "Adaptive EDCF: Enhanced service differentiation for IEEE 802.11 wireless ad-hoc networks," in *Proc. IEEE WCNC*, Mar. 2003, vol. 2, pp. 1373–1378.
- [14] H. Zhu, G. Cao, A. Yener, and A. D. Mathias, "EDCF-DM: A novel enhanced distributed coordination function for wireless ad hoc networks," in *Proc. IEEE ICC*, Jun. 2004, vol. 7, pp. 3886–3890.
- [15] X. Chen, H. Zhai, and Y. Fang, "Supporting QoS in IEEE 802.11e wireless LANs," *IEEE Trans. Wireless Commun.*, vol. 5, no. 8, pp. 2217–2227, Aug. 2006.
- [16] J. L. Sobrinho and A. S. Krishnakumar, "Quality-of-service in ad hoc carrier sense multiple access networks," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 8, pp. 1353–1368, Aug. 1999.
- [17] Z. J. Haas and J. Deng, "Dual busy tone multiple access (DBTMA)—A multiple access control scheme for ad hoc networks," *IEEE Trans. Commun.*, vol. 50, no. 6, pp. 975–985, Jun. 2002.
- [18] S. Wu, Y. Tseng, and J. Sheu, "Intelligent medium access for mobile ad hoc networks with busy tones and power control," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 9, pp. 1647–1657, Sep. 2000.
- [19] A. Veres, A. T. Campbell, M. Barry, and L. Sun, "Supporting service differentiation in wireless packet networks using distributed control," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 10, pp. 2081–2093, Oct. 2001.

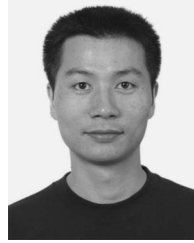
- [20] X. Yang and N. Vaidya, "Priority scheduling in wireless ad hoc networks," *Wireless Netw.*, vol. 12, no. 3, pp. 273–286, Jun. 2006.
- [21] C. E. Koksal, H. Kassab, and H. Balakrishnan, "An analysis of short-term fairness in wireless media access protocols," *ACM Sigmetrics*, vol. 28, no. 1, pp. 118–119, Jun. 2000.
- [22] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.
- [23] 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. IEEE INFOCOM*, Jun. 2002, vol. 2, pp. 599–607.



Sunmyeng Kim received the B.S., M.S., and Ph.D. degrees in information and communication from Ajou University, Suwon, Korea, in 2000, 2002, and 2006, respectively.

From May 2006 to February 2008, he was a Post-doctoral Researcher in electrical and computer engineering with the University of Florida, Gainesville. In March 2008, he then joined the School of Computer and Software Engineering, Kumoh National Institute of Technology, Gumi, Korea, as a Full-Time Lecturer. His research interests include re-

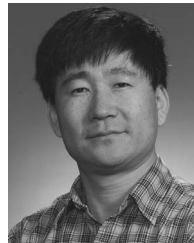
source management, wireless LANs and PANS, wireless mesh networks, and quality-of-service enhancement.



Rongsheng Huang (S'07) received the B.S. and M.S. degrees in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 1996 and 1999, respectively. He is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Florida, Gainesville.

From 1999 to 2001, he was with Huawei Technologies Co. Ltd., Shenzhen, China, as an R&D Engineer, working on GPRS and 3G projects. From 2002 to 2005, he was with UTStarcom Research

Center, Shenzhen, as a Senior Engineer and Team Leader on a 3G project. His research interests include media access control, protocol, and architecture for wireless networks.



Yuguang Fang (S'92–M'93–SM'99–F'08) received the Ph.D. degree in systems engineering from Case Western Reserve University, Cleveland, OH, in 1994 and the Ph.D. degree in electrical engineering from Boston University, Boston, MA, in 1997.

From July 1998 to May 2000, he was an Assistant Professor with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark. In May 2000, he joined the Department of Electrical and Computer Engineering, University of Florida, Gainesville, as an Assistant

Professor. He was promoted to Associate Professor in August 2003 and to Professor in August 2005. He has authored more than 200 papers in refereed professional journals and conference proceedings.

Prof. Fang has served on several editorial boards of technical journals, including the *IEEE TRANSACTIONS ON COMMUNICATIONS*, the *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS*, the *IEEE TRANSACTIONS ON MOBILE COMPUTING*, and *ACM Wireless Networks*. He was the recipient of the U.S. National Science Foundation Faculty Early Career Award in 2001 and the U.S. Office of Naval Research Young Investigator Award in 2002.