UPCF: A New Point Coordination Function With QoS and Power Management for Multimedia Over Wireless LANs

Zi-Tsan Chou, *Member, IEEE*, Ching-Chi Hsu, and Shin-Neng Hsu

***Abstract*—In this paper, we propose a new novel polling-based medium access control protocol, named UPCF (Unified Point Coordination Function), to provide power conservation and quality-of-service (QoS) guarantees for multimedia applications over wireless local area networks. Specifically, UPCF has the following attractive features. First, it supports multiple priority levels and guarantees that high-priority stations always join the polling list earlier than low-priority stations. Second, it provides fast reservation scheme such that associated stations with real-time traffic can get on the polling list in bounded time. Third, it employs dynamic channel time allocation scheme to support CBR/VBR transportation and provide per-flow probabilistic bandwidth assurance. Fourth, it employs the power management techniques to let mobile stations save as much energy as possible. Fifth, it adopts the mobile-assisted admission control technique such that the point coordinator can admit as many newly flows as possible while not violating QoS guarantees made to already-admitted flows. The performance of UPCF is evaluated through both analysis and simulations. Simulation results do confirm that, as compared with the PCF in IEEE 802.11, UPCF not only provides higher goodput and energy throughput, but also achieves lower power consumption and frame loss due to delay expiry. Last but not least, we expect that UPCF can pass the current Wi-Fi certification and may coexist with the upcoming IEEE 802.11e standard.**

***Index Terms*—IEEE 802.11, medium access control (MAC), multimedia, point coordination function (PCF), power management, quality of service (QoS).**

I. Introduction

WITH the proliferation of mobile devices and the advance of channel modulation technologies, there has been growing interest in providing quality-of-service (QoS) guarantees for multimedia applications over wireless local area networks (WLANs). A WLAN typically consists of a central base station, also known as an *access point* (AP), and a finite set of associated mobile stations. Since mobile stations are often operated by batteries or other exhaustible means for their energy, it is vital to incorporate *power saving* (PS) mechanisms into the design of wireless network protocols. Currently, IEEE 802.11 [13] is the most popular international *medium access control* (MAC) standard for WLANs. In the literature, there has been extensive work on improving IEEE 802.11 for prioritization, higher throughput, lower mean access delay, traffic scheduling, and power saving support. However, these issues are treated separately by different researchers and their solutions may not combine well with each other. Clearly, how to deliver all these functions in an integrated MAC scheme indeed poses a great challenge.

A. Related Work

The IEEE 802.11 defines two modes of operation: the *distributed coordination function* (DCF) and the *point coordination function* (PCF). The DCF used in the *contention period* (CP) employs carrier sense multiple access with collision avoidance (CSMA/CA) strategy to provide asynchronous best-effort service. The PCF used in the *contention-free period* (CFP) employs the polling strategy to provide isochronous connection-oriented service. PCF uses a *point coordinator* (PC), which should operate at the AP, to determine which station on the *polling list* currently has the right to transmit. When a PC is operating in a WLAN, the two coordination functions alternate, with a CFP followed by a CP, which are together referred to as a *CFP repetition interval* or a *superframe*. For a more complete and detailed presentation, please refer to the IEEE 802.11 specification [13]. One of the advantages of the alternating period approach is that even if the AP/PC fails, the entire MAC system is still operative since it can be switched to the *ad hoc network* configuration. However, there are several problems with PCF that make it less attractive for QoS and power conservation.

1) Any associated station intending to join the polling list should first send the reassociation frame to the AP during the CP. Since DCF is governed by a contention-based protocol, the reassociation frames need to compete with all other stations in the same cell, resulting in an unbounded reassociation delay. Hence, a real-time station with bad luck may never obtain the contention-free service.
2) IEEE 802.11 does not support the concept of differentiating frames with different user priorities [6]. The DCF is basically supposed to provide a long-term fair channel access to all contending stations in a distributed way. This implies that low-priority stations may join the polling list earlier and faster than high-priority stations.
3) In an infrastructure WLAN, IEEE 802.11 does not allow a station to send frames directly to any other stations within the same cell, and instead the AP should relay the frames always [6]. In this way, the channel bandwidth is indeed

Manuscript received July 13, 2003; revised August 9, 2004 and August 1, 2005; approved by IEEE/ACM Transactions on Networking Editor M. Krunz. This work was supported in part by the Innovative and Prospective Technologies Project (Project Number: 93-EC-17-A-99-R1-0461) of the Institute for Information Industry and sponsored by MoEA, Taiwan.

Z.-T. Chou is with the Networks and Multimedia Institute, Institute for Information Industry, Taipei 106, Taiwan, R.O.C. (e-mail: ztchou@nmi.iii.org.tw).

C.-C. Hsu is with the Institute for Information Industry, Taipei 106, Taiwan, R.O.C. (e-mail: cchsu@iii.org.tw).

S.-N. Hsu is with the Taiwan Semiconductor Manufacturing Company, Hsinchu, Taiwan, R.O.C. (e-mail: snshiu@tsmc.com.tw).

Digital Object Identifier 10.1109/TNET.2006.880170

consumed twice than directional communication between stations.

4) During the CFP, the medium occupancy time or the transmission time of a polled station is unpredictable and unrestrained. Any polled station is allowed to send a single frame that may be of an arbitrary length, up to the maximum of 2304 bytes (or 2312 bytes when the frame body is encrypted using WEP [13]). This may adversely degrade and ruin the performance of the other stations on the polling list.
5) Since PCF does not perform any admission control, the PC may admit a large number of real-time stations. Under the circumstances, several admitted stations may not receive the data or poll from the PC during the entire CFP, hence incurring unnecessary awakeness and energy expense.
6) When admitted stations desire to leave the polling list, they shall reassociate with the AP via DCF. The station without additional buffered data but having no chance to get off the polling list will response a *Null* frame when polled by the PC. These Null frames are simply the wastage of bandwidth, thus causing the PCF performance down.

To reduce the overhead of polling frames in PCF, the authors in [10] proposed the SuperPoll protocol. Instead of polling each station individually, in SuperPoll, the PC broadcasts a *superpoll* frame which contains the list of stations to be polled after sending the beacon. After receiving the superpoll, each station on the polling list can transmit the data frame in turn, according to the polling order, to any other station within the same cell. However, if a polled station does not hear its predecessor's transmission, then that station shall wait for the time interval allocated to it. This approach implies that, in CFP, the data frame length must be *fixed*. On the other hand, in SuperPoll, the PC will broadcast the CF-End to reset the *network allocation vector* (NAV) either after it receives the transmission from the last station on the polling list or until the CFPMaxDuration expires. Thus, SuperPoll may encounter the *idling-CFP accident*: Once the PC has successfully sent the beacon frame to set the NAV to lock out DCF-based access, but the superpoll frame is destroyed due to interference, then the entire CFP will be nearly idle and completely wasted. This is because, after broadcasting the superpoll, the PC is *not* allowed to get involved in the PCF operation any longer until the time to send the CF-End. The CF-Multipoll protocol [9] that was once considered in 802.11e is similar to the SuperPoll, except that it introduces a TXOP (*transmission opportunity*) field in the CF-multipoll frame to remove the restriction of fixed data frame length. The authors in [17] presented a traffic scheduling and an admission control scheme based on the CP-Multipoll protocol. Although these polling protocols [9], [10], [17] claim they can support peer-to-peer communications during the CFP, yet they may incur the *PS-induced frame loss problem*: Since the superpoll, CF-multipoll, and CP-multipoll frames do not specify the *receiver* for each transmission, if a polled station directly sends a data frame to the station currently in the *doze* state, then that data frame is certainly lost.

MAC protocols designed for QoS support should provide prioritization schemes since we do not hope that high-priority stations must contend fairly with low-priority stations for acquiring the medium access floor. The common idea behind the prioritization techniques in [4], [8], [14], [23] is to let a higher-priority frame have a shorter waiting time *during the CP*. In IEEE 802.11, prioritized access for different frame types is controlled through the use of different *interframe spaces* (IFSs), including SIFS (Shortest IFS), PIFS (Priority IFS), and DIFS (Distributed IFS). To accommodate additional QoS provision, IEEE 802.11e draft [14] proposes a new coordination function, called HCF (*hybrid coordination function*). The HCF defines two channel access mechanisms: EDCA (*enhanced distributed channel access*) and HCCA (*HCF controlled channel access*). Especially, the EDCA introduces a new type of IFS, named AIFS (*Arbitration IFS*), for different *access categories*. However, [19] pointed out that, since the new AIFS values are not shorter than DIFS, the frame of a station using the existing DCF *may* take priority over that of a station using EDCA. The authors in [8] proposed the *contention window separation* scheme such that a higher-priority frame has a shorter backoff time. EDCA [14] and [4] proposed the *contention window differentiation* scheme such that the minimum and maximum values of the contention windows (CWs) of a high-priority frame are respectively smaller than those of a low-priority frame. Although these schemes [4], [8], [14] can provide differentiated services, yet [19], [22] pointed out that they may suffer from the *priority reversal problem*: Since the number of *random* backoff slots is associated with the CW and the CW is exponentially proportional to the number of retransmission attempts, a high-priority *backlogged* frame may experience a longer backoff time than a low-priority *unbacklogged* frame. Note that a frame which involved in a collision and must be retransmitted is said to be *backlogged*[5]. Although various *black-burst* (BB) contention-based prioritization schemes proposed in [3], [22], [25] can eliminate the priority reversal problem in a single-hop ad hoc network, [23] indicated that the BB contention is not a regular mechanism defined in the IEEE 802.11 standard, thus it is difficult to be overlaid on the current CSMA implementations.

MAC protocols designed for QoS support had better provide time-bounded reservation schemes since we hope that real-time stations can speedily reserve the contention-free periodic access right. IEEE 802.11 and 802.11e adopt the DCF and EDCA respectively as the reservation schemes. In the DBASE protocol [23], real-time stations employ the p-persistent backoff scheme to compete for joining the reservation list during the time interval between PIFS and DIFS. DBASE assumes a tiny *constant* contention window (3 slots) for real-time traffic and a huge period of $\mathrm{DIFS} = \mathrm{SIFS} + 5 \times \mathrm{SlotTime}$ for non-real-time traffic, which may degrade the channel utilization. The authors in [22] modify the randomized initialization protocol [18] to resolve reservation contention based on unbiased coin-flipping. However, due to the nature of randomness, these contention-based schemes [13], [14], [22], [23] cannot guarantee bounded reservation time. The simplest way to offer the time-bounded reservation service is *roll-call polling* [16]; in other words, the AP polls every station in sequence and check whether it has data to transmit. However, since the AP polls every station, it may happen that many stations are polled only to learn that they have nothing to send, thus unnecessarily delaying the stations with packets. The STRP protocol presented in [21] utilizes the capture effect to improve the roll-call polling. In STRP, the associated stations are split by the PC into two logical rings, the active ring and idle ring, according to whether they have pending data. The PC makes use of the *query/transmit*-poll to enable a station in the active ring to transmit a data frame by a stronger power, while inviting an active station in the idle ring to simultaneously response a lasting jamming signal by a weaker power. Once the jamming signal is detected by the PC, that station in the idle ring will be shifted to the active ring. Although this approach may shorten the reservation time, yet once the active ring becomes empty, STRP is reduced to roll-call polling. Even worse, STRP may suffer from the near-far problem or costly dual transceivers. The CARMA-NTQ protocol [11] proposed for single-hop ad hoc networks employs the deterministic first-success tree-splitting algorithm to efficiently resolve collisions in bounded time, thus achieving high channel utilization even at high load. However, these reservation-based protocols [11], [21]–[23] do not take power conservation into consideration.

In 802.11, the PCF uses the *traffic indication map* (TIM) to inform which stations cannot doze off during the CFP even though the PC may unfortunately not poll them. To improve the power management mechanisms used in PCF, the authors in [26] proposed three alternative directory protocols that can be used by the PC to schedule the transmission of data and the dozing of stations. However, due to the lack of TXOP fields in the directory structures, their protocols [26] require that every data frame transmitting in the CFP must be fixed-sized.

B. Our Contributions

So far there have been some commercial MAC controller supporting both DCF and PCF, such as HelloWLAN [12] by HelloSoft, VT6655 [28] by VIA Networking, and WL6000 [30] by Duolog. However, all the above mentioned challenges provide solid motivations for the need of redesigning PCF. Accordingly, we will tailor the PCF mechanisms such that our new protocol can *coexist* with the DCF, while providing power conservation and QoS guarantees to real-time multimedia applications. We name the resulting protocol UPCF (*Unified Point Coordination Function*). The characteristics of UPCF are as follows.

1) UPCF adopts the *handshaking* technique, instead of using backoff or BB mechanisms, to implement traffic prioritization *during the CFP*. Above all, UPCF guarantees that high-priority stations are always admitted to the polling list earlier than low-priority stations.
2) UPCF adopts the deterministic tree-splitting algorithm as the reservation mechanism such that associated stations with real-time traffic can get on the polling list in *bounded* time without relying on the reassociation. In addition, UPCF uses the *piggyback* technique such that admitted stations can get off the polling list easily and quickly without performing a reassociation.
3) UPCF employs the elaborate V-POLL (*vector-list poll*) frame to support both uplink/downlink and peer-to-peer communications in the CFP without suffering the PS-induced frame loss. Importantly, the PC in UPCF is still able to retain control of the medium, when a polled station does not respond, without leaving the medium idle for more than a *PIFS* period. By this way, we ensure that the idling-CFP accident will never occur.
4) With dynamic TXOP allocation scheme, UPCF provides *isolation* among admitted multimedia flows while utilizing bandwidth resources as efficiently as possible. Specifically, our TXOP allocation scheme can be regarded as an enhancement of DBASE [23] in that UPCF is capable of offering *per-flow* probabilistic bandwidth assurance.
5) Since the length of the maximum CFP duration is limited, we integrate the *run-time admission control* mechanism into the registration process such that the PC can admit as many newly arriving flows as possible while maintaining QoS guarantees made to already-admitted flows.
6) UPCF achieves power conservation via the following three approaches. First, in contrast with contention-based MAC protocols, UPCF adopts the reservation and polling-based access scheme to reduce energy waste on collisions and retransmissions as far as possible. Second, UPCF utilizes the V-POLL frame to let PS stations which cannot partake in the polling activity immediately return to the doze state. Last, UPCF employs the power-conserving scheduling such that PS stations which will partake in the polling activity can spend as little awake time as possible.
7) In 802.11e draft [14], the EDCA is used only *in the CP* and the HCCA can be used in both CP and CFP. In addition to PCF, HCCA further allows the PC to flexibly poll a real-time station with granted TXOP *in the CP*. Importantly, being *independent* of any channel access mechanisms operating in the CP, UPCF can pass the current Wi-Fi certification [13, pp. 245–247], [29] and may coexist with the upcoming IEEE 802.11e standard. Table I summarizes and compares the yet-to-be-presented UPCF protocol with several representative MAC protocols mentioned in the previous subsection.

II. The UPCF Protocol

A. Network Model and Assumptions

The basic building block of the IEEE 802.11 network is the *cell*, also known as the *basic service set* (BSS). A BSS is composed of an AP and a finite set of mobile stations. The typical diameter of the basic service area (BSA) of a BSS is considered only on the order of 100 feet [23]. Therefore, we assume that all stations within the same BSS are able to communicate to each other directly. In IEEE 802.11 [13], a station should first authenticate and associate with *an* AP (or reassociate with a new AP) to become a member of an infrastructure BSS. When the association request is granted, the AP responds with a status code of 0 (successful) and the *Association ID* (AID). The AID is an integer identifier used to logically identify the mobile station. The AP/PC can thus maintain a list of finite stations associated within its BSS and updates it whenever a new station joins or a station leaves the BSS. However, UPCF disables the *CF-Pollable* and *CF-Poll Request* subfields of the *capacity information* field in (re)association request frames. Instead, UPCF provides a new reservation mechanism to let real-time stations get on/off the polling list quickly without relying on the reassociation.

TABLE I
COMPARISON OF DIFFERENT MAC PROTOCOLS FOR QOS AND POWER CONSERVATION IN WLANS

Protocol	Multi-Level Prioritization	Priority-Reversal Possibility	Time-Bounded Reservation	Bandwidth Allocation	Admission Control	Power Management
802.11 standard [13]	no	×	no	no	no	yes
802.11e draft [14]	yes	yes	no	yes	yes	yes
Backoff schemes [4], [8]	yes	yes	no	no	no	no
BB schemes [3], [22], [25]	yes	no	no	no	no	no
CP-Multipoll [17]	no	×	no	yes	yes	no
DBASE [23]	two*	no	no	yes	yes	no
STRP [21]	no	×	yes	no	no	no
CARMA-NTQ [11]	no	×	yes	no	yes	no
Stine and Veciana [26]	no	×	no	no	no	yes
UPCF	yes	no	yes	yes	yes	yes

*DBASE is unable to support more than *two* traffic priority levels.

B. CFP Structure and Timing

The UPCF mechanism in the MAC layer architecture is built on top of the CSMA/CA-based DCF to provide prioritized and parameterized QoS services. The DCF and UPCF can coexist in a manner that permits both to operate concurrently within the same BSS. In a BSS, the PC takes charge of channel time allocation and makes these two coordination functions alternative, with a CFP (during which UPCF is active) followed by a CP (during which DCF is active), which are together referred as a *superframe*. A mobile station can operate in either the *active* mode or the *PS* mode. At the nominal start of each CFP, known as the TBTT (*target beacon transmission time*), every PS station shall wake up and remain active to listen for the V-POLL frame; meanwhile, the PC continuously monitors the channel and then seizes its control by transmitting a *beacon* frame after the *PIFS* medium idle time. One component of the beacon announcement is the maximum duration of the CFP, *CFPMaxDuration*. Every station receiving the beacon shall update its NAV to the CFPMaxDuration. This NAV is used for preventing a DCF-based station from taking control of the medium during the CFP. In UPCF, as depicted in Fig. 1, the CFP is further divided into three periods: the *prioritization period*, the *collision resolution period*, and the *polling period*. The first two periods are together called the *registration period*. During the prioritization period, the PC performs a series of handshakes to guarantee that high-priority stations are always admitted to the polling list earlier than low-priority stations. During the collision resolution period, the PC performs a deterministic tree-splitting algorithm to probe which stations undergo the prioritization period desire to join the polling list. Once the registration process terminates, the PC broadcasts a V-POLL frame to announce the start of the polling period. The V-POLL frame contains a list of vectors and each vector is composed of a yet-to-be-polled station, its intended receiver, and the granted TXOP. Upon examining the V-POLL frame, a PS station that can be neither a sender nor a receiver during the polling period may reenter to the doze state. Note that, in the doze state, a station is unable to transmit or receive but consumes very low power. After the end of the polling period, the PC broadcasts the *CF-End* frame to let all stations reset their NAV and enter the CP.

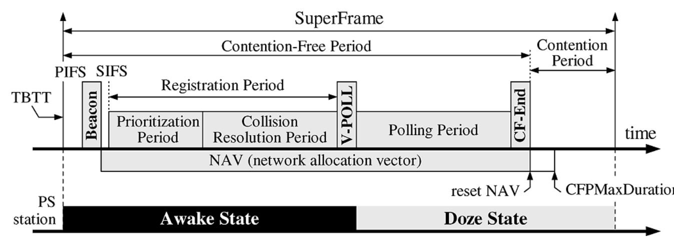


Fig. 1. Superframe structure and the awake/doze state transition of a PS station.

Consistent with the IEEE 802.11 [13], the minimum length of the CP, $\mathrm{CP}_{\min}$, is the time required to transmit and acknowledge one maximum-sized MPDU (*MAC protocol data unit*); namely, $\mathrm{CP}_{\min} = \mathrm{DIFS} + T_{\max \mathrm{MPDU}} + \mathrm{SIFS} + T_{\mathrm{ACK}}$, where T_{ACK} is the time needed for sending the ACK frame. The value of CFPMaxDuration shall be limited to allow coexistence between DCF and UPCF traffic. Thus, we have $\mathrm{CFPMaxDuration} = \mathrm{SF} - \mathrm{CP}_{\min}$, where SF is the length of the superframe. Since the length of CFPMaxDuration is limited, the overrun of the registration process may shorten the polling period, violating the quality of already-admitted connections. Hence, a run-time admission control is established to assist the PC in determining when the registration period shall be terminated. In particular, when the polling list size reaches the saturation point (see Section II-G), the PC may directly dive into polling period at the start of the CFP without first performing the registration procedure. A surprising phenomenon in UPCF is that collisions may occur in the CFP. However, during the entire CFP, associated stations can transmit frames only when they are allowed to do so by the PC. Consequently, the PC can control these collisions effectively and without chaotic events.

C. Prioritization Procedure

In UPCF, priority levels are numbered from 0 to H, with H denoting the highest priority level. A frame with priority 0 shall be sent via the DCF. On the other hand, only the active real-time station that has a flow with priority level ranging from 1 to H can participate in the registration process. Note that a real-time station is *active* if it desires to get on the polling list. Besides, a *flow* is a continuous stream of frames that have the same source, destination, priority level, and quality of service.

After broadcasting a beacon and waiting for a SIFS period, the PC sends the PE_H (*priority enquiry*) frame to invite every active station whose priority level equals H to reply the PR (*priority response*) frame. On receiving the PE_H frame, an active station with priority level H shall acknowledge a PR frame after a SIFS period. At the end of the handshake, the PC can obtain the ternary feedback information according to stations' responses: (i) IDLE: The PC does not receive any PR frames. (ii) SINGLE: The PC successfully receives a single PR frame which includes the *sender AID* and the *destination MAC address*. That AID will be placed on the polling list. (iii) COLLISION[1]: This event occurs if the outcome of the handshake is neither IDLE nor SINGLE.

If the conclusion of the current handshake is IDLE (SINGLE, respectively), the PC may proceed to the next handshake by issuing the PE_{H-1} frame after an elapsed PIFS (SIFS, respectively). The priority probing process keeps running until the occurrence of a COLLISION, the delivery of the PE_1 frame, or a failure in the run-time admission test (see Section II-G), whichever comes first. Especially, once the PC recognizes a COLLISION event, it will send an RE (*registration enquiry*) frame to announce the start of the collision resolution period. During the collision resolution period, the PC executes the deterministic collision resolution procedure to discover which active stations bring the COLLISION event. The prioritization operation is essentially that of polling, with the PC polling each of the priority groups in a descending order. The average overhead of the prioritization operation is expected to be low since the value of H is usually small. (Generally, $H \leq 7$. In IEEE 802.11e, $H = 2$; that is, only two access categories (*voice* and *video*) take priority over DCF traffic.) It is noteworthy that a lower-priority station will be blocked if it has no chance to send out a PR frame during the entire prioritization period. We could adopt the *aging* policy [3], [24] (As time progresses, so does the priority of the flow.) to conquer the problem of indefinite blockage or starvation.

The illustration in Fig. 2 shows how the prioritization procedure works. From Fig. 2, we can observe that, in UPCF, the synchronization among the PC and associated stations is well controlled by using different interframe spaces. Actually, all UPCF transmissions during the CFP are separated only by *SIFS* or *PIFS*. This mechanism ensure that even though the beacon frame is lost, the PC can still safeguard its control of the medium against the DCF-based interference. Consistent with the IEEE 802.11, we let $\mathrm{PIFS} = \mathrm{SIFS} + \mathrm{SlotTime}$ and $\mathrm{DIFS} = \mathrm{SIFS} + 2 \times \mathrm{SlotTime}$. As per IEEE 802.11, the SIFS interval is equal to the sum of receiver radio frequency delay, receiver PLCP (physical layer convergence procedure) delay, the MAC processing delay, and the transceiver turnaround time. The SlotTime accounts for the carrier sensing time, the transceiver turnaround time, the MAC processing delay, and the air propagation delay.

[1]Misinterpreting a SINGLE handshake result as a COLLISION one due to noise errors in the registration period *may* result in, at most, two additional handshakes, the penalty of which is $2\times T_{\mathrm{RE}}+T_{\mathrm{RR}}+\mathrm{PIFS}+2\times\mathrm{SIFS}$. After reading Sections II-D and II-G, the reader will understand that, even in an error-prone channel, the length of the registration period can be still well controlled by the run-time admission control algorithm.

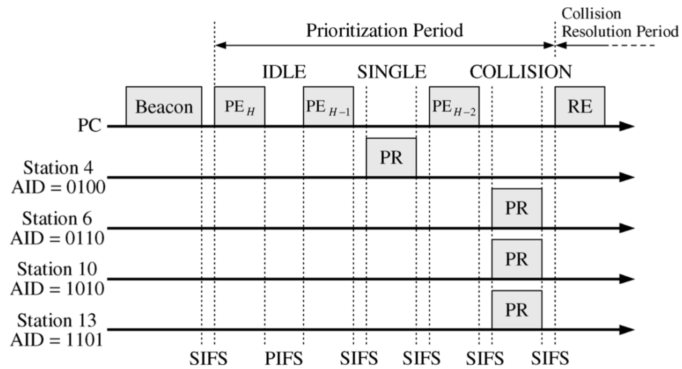


Fig. 2. An example of the prioritization procedure. We assume that there are 15 associated stations in a BSS. Stations 4, 6, 10, and 13 intend to join the polling list. In the first round, the PC sends the PE_H frame and no one responds. In the second round, only station 4 replies the PR frame and joins the polling list successfully. At the end of the third round (handshake), the PC perceives a COLLISION event and then performs a collision resolution procedure.

D. Collision Resolution Procedure

Theoretically, any collision resolution multi-access algorithms are applicable in the collision resolution period. The reasons for choosing the *identifier*-based tree-splitting algorithm [5] are due to its simplicity, stability, and *bounded* collision resolution period. The basic idea of the tree-splitting algorithm is to implement a *preorder traversal* of the *splitting tree*. Specifically, when a COLLISION occurs, the PC splits the set $\mathcal{A}$ of stations involved in the collision into two subsets, $\mathcal{A}_1$ and $\mathcal{A}_2$. The PC first recursively resolves the collision of $\mathcal{A}_1$, and then resolves the collision of $\mathcal{A}_2$ independently. We assume that the close of the prioritization period results from the transmission of multiple PR_h frames, where $1 \leq h \leq H$. During the collision resolution period, the PC sends the RE (*registration enquiry*) frame which contains the value of h and the set of binary strings (AddressPattern $\mathcal{A}$) to invite active stations to reply the RR (*registration response*) frame. Upon reception of the $\mathrm{RE}(h, \mathcal{A})$ frame, the active station with priority level h and $\mathrm{AID} \in \mathcal{A}$ shall acknowledge an RR frame after a SIFS period. At the end of the handshake, the PC adjusts the value of $\mathcal{A}$ according to stations' responses (SINGLE/IDLE/COLLISION). Especially, if the PC successfully receives a single RR frame which includes the sender AID and the destination MAC address, then the PC will add that AID to the polling list. This AID probing process keeps running until the completion of the tree traversal or a failure in run-time admission test (see Section II-G), whichever comes first.

Continuing the example in Fig. 2, Fig. 3 shows how the deterministic tree-splitting procedure works. In the first round, the PC sends out the RE frame with $\mathcal{A} = \{***0\}$, asking for responses. Since both the station 6 ($\mathrm{AID} = 0110$) and station 10 ($\mathrm{AID} = 1010$) belong to the set $\mathcal{A}$, then they reply, and their replies collide. Upon recognizing the COLLISION event, the PC halves the range of $\mathcal{A}$ ($\mathcal{A} = \{**00\}$) and enquires again. This time, the PC will discover an IDLE event. However, it is pointless for the PC to further probe the range $\mathcal{A} = \{**10\}$ since it is predictable to have a COLLISION. At the end of the third round (handshake), the PC correctly receives a single RR frame which contains the $\mathrm{AID} = 10$ of the sender. The PC then

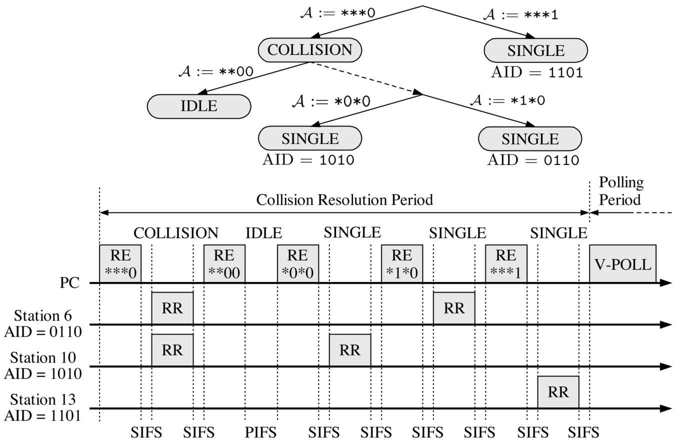


Fig. 3. An example of the collision resolution procedure. The tree structure represents a particular pattern of IDLEs, SINGLEs, COLLISIONs resulting from a sequence of splitting. We can observe that UPCF utilizes different interframe spaces to realize synchronous operations and hence has no need for time slotting as prior MAC protocols based on collision resolution do [5], [16], [18].

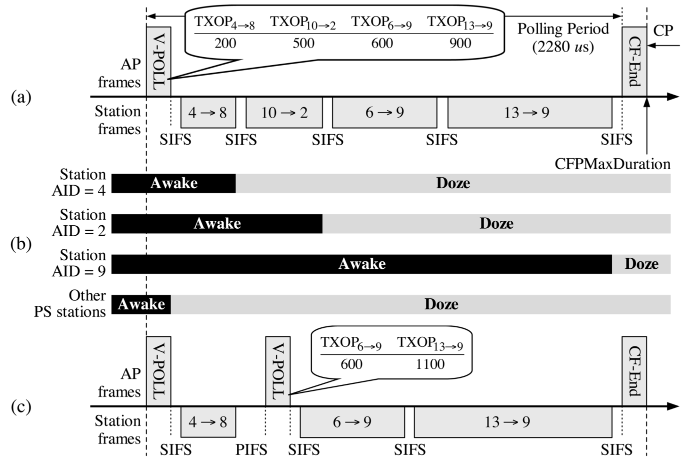


Fig. 4. We assume that, during the registration period, stations 4, 10, 6, and 13 declare $(\mathcal{D}_4 = 200, \mathcal{G}_4 = 200)$, $(\mathcal{D}_{10} = 600, \mathcal{G}_{10} = 400)$, $(\mathcal{D}_6 = 600, \mathcal{G}_6 = 900)$, and $(\mathcal{D}_{13} = 1100, \mathcal{G}_{13} = 700)$, respectively. Part (a) shows that the PC uses a single V-POLL frame to specify the access order and the TXOP for each polled station; namely, $\langle(4,8,200),(10,2,500),(6,9,600),(13,9,900)\rangle$. Note that the aggregate TXOP of station 9 is $600 + 900 = 1500$. The PC polls the station 4 first because it has the smallest aggregate TXOP (200). Part (b) shows the power management operation in UPCF. Part (c) shows that the PC seizes the medium by rebroadcasting the V-POLL frame when it does not receive a response from station 10 after an elapsed PIFS. Note that, in this case, the TXOP demand of station 13 is luckily satisfied.

places that $\mathrm{AID} = 10$ on the polling list. Continuing in this manner, the PC can skip over large chunks of the address space that have no active stations. However, when all stations are active and have the same priority level, this will result in doubling the number of RE polls, as compared with roll-call polling. Fortunately, once active stations get on the polling list, they can reserve the periodic access right and will not participate in the registration process again. In particular, when contending traffic is not heavy, tree-splitting algorithm is quite efficient [5], [11], [16]. In sum, the tree-splitting operation is essentially that of polling, with the PC systemically and adaptively controlling the number of allowably contending stations to finally identify each active station. From Fig. 3, we can observe that after five handshakes, stations 6, 10, and 13 join the polling list; and then the PC broadcasts a V-POLL frame to let each active station know whether it has been successfully placed on the polling list.

E. Polling Procedure and Power Management

At the beginning of the polling period, the PC sends the V-POLL (*vector-list poll*) frame to specify the access order, the receiver AID, and the medium occupancy time (TXOP) for each polled station. Fig. 5(b) presents the format of the V-POLL frame. Each flow in the polling list has its corresponding *poll record*. The Record Count field is set to the number of poll records. The TXOP (transmission opportunity) subfield specifies the time duration during which the polled station has the right initiate frame exchange sequences onto the wireless medium [14]. (We will describe how the PC dynamically assigns the value of TXOP for each admitted flow in the next subsection.) Via the V-POLL frame, UPCF supports both uplink/downlink and peer-to-peer communications in a WLAN. Clearly, if the PC has data to send, it can also add its ID to the polling list. Fig. 4(a) depicts how the polling procedure works and Fig. 5(c) shows the MPDU frame format used in the CFP. On inspecting the V-POLL frame, a PS station that can neither transmit nor receive data frames during the polling period may return to the doze state. By contrast, each polled station needs to keep track of the channel activity and automatically initiates its transmission a SIFS period after the end of the transmission of its predecessor in the polling order. To conserve energy, a PS station that will be a sender or receiver may remain in the awake state for only a portion of the polling period through the time that the PS station finishes sending or receiving data frames. Fig. 4(b) depicts the power management operation in the CFP. From Figs. 4 and 5(b), we can observe that, with the help of V-POLL frame, UPCF successfully eliminates the PS-induced frame loss problem (mentioned in Section I-A). To minimize the average time spent awake, the PC in UPCF adopts the *shortest-job-first* policy proposed in [26] to schedule the poll records, where the job size corresponds to the aggregate TXOP. More specifically, in order to put the most PS stations to sleep soonest, UPCF uses the following algorithm to schedule the polling order.

S1. Calculate the aggregate TXOP of each station that appears in the set of flows waiting to be scheduled. The *aggregate TXOP* of station k is defined as the sum of the TXOP values of the yet-to-be-scheduled flows where the sender or receiver is station k.

S2. Schedule those flows involving the station that has the smallest aggregate TXOP.

S3. Repeat steps **S1** and **S2** until all flows are scheduled.

During the polling period, the time gap between two successive transmissions is generally a SIFS period. However, as depicted in Fig. 4(c), if a polled station, say $\mathrm{AID} = j$, does not respond, then the PC resumes sending the V-POLL frame which contains the remaining untransmitted stations on the polling list after an elapsed *PIFS*. This permits the PC to retain control of the medium, thus stopping the idling-CFP accident from happening. It is noteworthy that, in UPCF, the PC will not remove that station j from the polling list until its no-response event

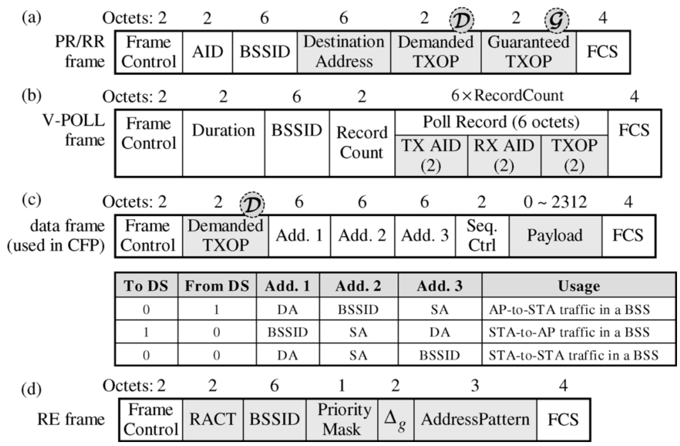


Fig. 5. The formats of UPCF frames.

has occurred for $\mathcal{K}$, say $\mathcal{K} = 3$, consecutive superframes. In IEEE 802.11, the PC has an obligation to acknowledge the receipt of data received from each polled station when performing the polling procedure. However, it is well known that real-time services, such as voice and video, can tolerate a small amount (1%∼3%) of dropped frames without suffering a large quality degradation [23]. Therefore, we adopt the *optional-ACK* policy for UPCF; in other words, each polled station shall explicitly indicate whether it requires an ACK when sending out an MPDU. In case the positive ACK is required but the MPDU was not properly received at the destination, the source station will be able to retransmit that MPDU when it is polled the next time. Fig. 4(a) shows an example that all real-time stations require no acknowledgements. Once an admitted station finishes sending its real-time flow at the present polling period and desires to tear down the connection, it shall set the *more data* bit to 0 in the *frame control* filed. When the PC receives this information, it will remove that station from the polling list. In this way, each admitted station can easily and quickly get off the polling list without performing a reassociation.

F. Dynamic TXOP Allocation Procedure

To receive performance assurance and make a reservation, an application (station) shall first characterize the traffic flow that it will inject into the WLAN and specify its desired TXOP (denoted by $\mathcal{G}$) that the PC must guarantee in each polling period. During the delivery of a continuous media stream, a real-time station may demand different TXOP (denoted by $\mathcal{D}$) in each polling period. Obviously, in case that $\mathcal{D} > \mathcal{G}$ happens, the PC may not be able to satisfy station's TXOP demand. Thus, for transporting the *variable bit rate* (VBR) traffic (e.g., video stream), an active real-time station shall estimate the value of $\mathcal{G}$ that satisfies the inequality $\Pr[\mathcal{D} \leq \mathcal{G}] \geq 1 - \varepsilon_{\mathrm{VBR}}$, where the value of $0 \leq \varepsilon_{\mathrm{VBR}} < 1$ reflects individual user's *tolerable TXOP-insufficiency probability*. For a VBR flow with mean μ and variance σ^2 of the bit rate, we could obtain

$$\mathcal{G} = \left(\mu + \sigma\sqrt{\frac{1-\varepsilon_{\mathrm{VBR}}}{\varepsilon_{\mathrm{VBR}}}}\right) \times \frac{\mathrm{SF}}{\mathrm{CDR}} \quad (1)$$

using the one-sided Chebyshev inequality [20], where SF is the length of the superframe and CDR is the channel data rate.[2] In [1], [22], [23], the authors assume that VBR video traffic follows the truncated exponential distribution with the minimum bit rate α, the peak bit rate β, and the mean bit rate μ. In this case, the value of $\mathcal{G}$ can be expressed as

$$\mathcal{G} = \left[\alpha - \gamma \ln\left(\varepsilon_{\mathrm{VBR}} + (1-\varepsilon_{\mathrm{VBR}})e^{(\alpha-\beta)/\gamma}\right)\right] \times \frac{\mathrm{SF}}{\mathrm{CDR}} \quad (2)$$

where γ is the solution of the following nonlinear equation:

$$\mu = \gamma + \frac{\alpha - \beta \times e^{(\alpha-\beta)/\gamma}}{1 - e^{(\alpha-\beta)/\gamma}}. \quad (3)$$

Derivations of (2) and (3) can be found in [7]. Note that the value of γ can be solved using numerical techniques. For example, the authors in [1], [22], [23] set $\alpha = 120$ K, $\beta = 420$ K, and $\mu = 240$ K; so we can get $\gamma \approx 244$ K. On the other hand, for transporting the *constant bit rate* (CBR) traffic (e.g., audio stream) with bit rate μ, we could assign

$$\mathcal{G} = \mu(1 - \varepsilon_{\mathrm{CBR}}) \times \frac{\mathrm{SF}}{\mathrm{CDR}} \quad (4)$$

where $0 \leq \varepsilon_{\mathrm{CBR}} < 1$ reflects individual user's *tolerable bandwidth loss ratio*. Expectably, the higher level of quality assurance the mobile user desires, the larger value of $\mathcal{G}$ the mobile station should request, while the more access fee the mobile user may be charged by the wireless network service provider. Accordingly, determining the appropriate value of $\mathcal{G}$ requires a trade-off among individual flow's TXOP assurance, individual user's access cost, and the whole network's channel utilization. In our simulations, we set the default values $\varepsilon_{\mathrm{VBR}} = 0.5$ and $0 \leq \varepsilon_{\mathrm{CBR}} < 0.03$.

In UPCF, an active real-time station shall declare its determined value of $\mathcal{G}$ using the PR or RR frame during the registration period. The PR/RR frame format is shown in Fig. 5(a). In PR/RR frame, the active real-time station uses the *guaranteed TXOP* field and the *demanded TXOP* field, respectively, to inform the PC the value of $\mathcal{G}$ and its TXOP demand $\mathcal{D}$ in the current CFP. Moreover, as shown in Fig. 5(c), each polled station piggybacks the *demanded TXOP* field with the MPDU to declare its required TXOP in the next polling period.

We now present how the PC allocates TXOP to provide isolation among admitted flows while utilizing channel time as efficiently as possible. The primary principle underlying the dynamic TXOP allocation procedure is that all flows with TXOP demands less than their declared amounts of guaranteed TXOP must be satisfied, while the unused CFP channel time will be allocated according to the weighted fair sharing scheme to the remaining flows with TXOP demands larger than their declared amounts of guaranteed TXOP. The TXOP allocation procedure operates formally as follows. Let $\mathcal{L} = \{A_1, A_2, \ldots, A_\ell\}$ be the polling list, where A_i denotes the station with $\mathrm{AID} = A_i$ and $\ell = |\mathcal{L}|$ equals the cardinality of the polling list. Let $\mathcal{G}_{A_i}$ and $\mathcal{D}_{A_i}$, respectively, be the guaranteed TXOP and demanded

[2]Note that CDR is *mobile-specific*. How to determine to the optimal CDR between the transmitter and the receiver is beyond the scope of this paper and the IEEE 802.11 specification [13]. It could be estimated by many methods, such as using GPS (Global Positioning System) or the measured signal strengths and signal-to-noise ratios from the history of received frames.

TXOP declared by station A_i. Before broadcasting the V-POLL frame, the PC calculates the value

$$\Upsilon = \text{CFPMaxDuration} - (T_s + \text{PIFS} + T_{\text{beacon}} + \text{SIFS} + T_{\text{reg}} + T_{\text{V-POLL}} + \text{SIFS} + T_{\text{CF-End}}) \quad (5)$$

where T_s is the length of the stretching time (see Section II-G) and T_{reg} is the length of the registration period. Note that the run-time admission control scheme described in Section II-G will ensure that each admitted station A_i can acquire the medium occupancy time at least $\min\{\mathcal{D}_{A_i}, \mathcal{G}_{A_i}\}$. Hence, the *residually sharable channel time* ($RSCT$) in the current CFP that can be fairly shared by those admitted stations whose demanded TXOP exceeds its declared guaranteed TXOP can be expressed as

$$RSCT = \Upsilon - \sum_{i=1}^{\ell} \Big(\min\{\mathcal{D}_{A_i}, \mathcal{G}_{A_i}\} + \text{SIFS} \Big). \quad (6)$$

The PC then allocates TXOP_{A_i} to station A_i according to the following formula:

$$\text{TXOP}_{A_i} = \begin{cases} \mathcal{D}_{A_i}, & \text{if } \mathcal{D}_{A_i} \le \mathcal{G}_{A_i}, \\ \min\left\{ \mathcal{D}_{A_i}, \mathcal{G}_{A_i} + \left\lfloor RSCT \times \frac{\mathcal{D}_{A_i} - \mathcal{G}_{A_i}}{\sum_{\mathcal{D}_{A_k} > \mathcal{G}_{A_k}} (\mathcal{D}_{A_k} - \mathcal{G}_{A_k})} \right\rfloor \right\}, \\ \text{if } \mathcal{D}_{A_i} > \mathcal{G}_{A_i}. \end{cases} \quad (7)$$

Take Fig. 4(a) for example, $\mathcal{L} = \{A_1 = 4, A_2 = 10, A_3 = 6, A_4 = 13\}$, CDR = 11 Mbps, $T_{\text{V-POLL}} = 30$ μs, $T_{\text{CF-End}} = 15$ μs, SIFS = 10 μs, and PIFS = 30 μs; according to (5)–(7), we have $RSCT = 300$ μs, $\text{TXOP}_4 = 200$ μs, $\text{TXOP}_6 = 600$ μs, $\text{TXOP}_{10} = 400 + \lfloor 300 \times 200/(400+200) \rfloor = 500$ μs, and $\text{TXOP}_{13} = 700 + \lfloor 300 \times 400/(400+200) \rfloor = 900$ μs.

It is noteworthy that once a polled station A_j does not respond in the polling period, the PC will recompute $RSCT$ and retransmit the V-POLL frame to announce the newly calculated TXOP values to the remaining members on polling list. Besides, in the next polling period, the PC will reserve $\text{TXOP}_{A_j} = \mathcal{G}_{A_j}$ for that station A_j since its demanded TXOP is unknown while its QoS requirement still needs to be guaranteed. Consider the example shown in Fig. 4(c), since no response is heard from station 10, the PC rebroadcasts the V-POLL frame after an elapsed PIFS and reports that $\text{TXOP}_6 = 600$ μs and $\text{TXOP}_{13} = 1100$ μs. Further, the PC will allocate $\text{TXOP}_{10} = 400$ μs for station 10 in the next polling period.

G. Run-Time Admission Control

Since the length of CFPMaxDuration is limited, the PC shall persist in monitoring the channel time usage and determine when to terminate the registration process in order not to violate TXOP guarantees made to already admitted stations. Conventional admission control schemes [16], [17] require that the mobile user submits its QoS requirement when making a reservation, and then the PC executes the admission test to decide whether to accept/reject that connection request

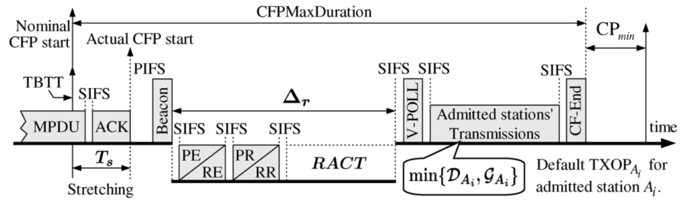


Fig. 6. Run-time admission control process and timing relationship between $RACT$ and Δ_r.

according to available resources. However, such a traditional approach is not suitable for UPCF in that the reservation request/response frame exchange failing the admission test simply wastes the scarce radio bandwidth. Instead, UPCF adopts the *mobile-assisted* admission control scheme: During the registration period, the PC evaluates the channel time usage based on the default medium occupancy time reserved for admitted flows, and then piggybacks the available channel time information with the PE/RE frame. Upon receipt of the PE/RE frame, active real-time stations take the admission test and check whether the remaining available channel time (RACT) is sufficient to meet their QoS needs. Those who pass the admission test can reply the PR/RR frames and report their QoS requirements; while those who fail the admission test shall abort the contention in the remaining registration period and wait for the next CFP. A valuable by-product of this approach is that contending registration traffic may be further reduced, making the tree-splitting algorithm more efficient. The PC then decides whether to proceed to the next PE/PR or RE/RR handshake according to the revised RACT and information collected from the received PR/RR frame(s). Importantly, the following two principles guide the design of UPCF admission control algorithm.

P1. The PC must make sure that the progress of the registration process will not affect the default medium occupancy time, $\min\{\mathcal{D}_{A_i}, \mathcal{G}_{A_i}\}$, of each admitted station $A_i \in \mathcal{L}$ on the polling list. Recall that, after the end of the registration period, the PC will calculate the newly appropriate TXOP values for all admitted stations via (5)–(7).

P2. It is possible for contention-based service runs past the nominal start of the CFP (TBTT). As per IEEE 802.11 [13], in the case of a busy medium due to DCF traffic, the CFP is *foreshortened* and the beacon should be delayed for the time required to complete the existing DCF frame exchange. Such a phenomenon is called *stretching* and we depict the stretching event in Fig. 6. The length of the stretching time T_s may be up to $\widehat{T}_s = T_{\text{RTS}} + T_{\text{CTS}} + T_{\max \text{MPDU}} + T_{\text{ACK}} + 3 \times \text{SIFS}$. The PC must make sure that QoS requirements of any admitted station will be guaranteed during the *flow lifetime* even in the worst case scenario, that is, when $T_s = \widehat{T}_s$ and $\mathcal{D}_{A_i} \ge \mathcal{G}_{A_i}$ for all $A_i \in \mathcal{L}$.

We now introduce some notations used to facilitate the specific presentation of run-time admission control algorithm.

- Let O_{CFP} denote the fixed overhead in a CFP. If $\mathcal{L} \ne \emptyset$, then we have

$$O_{\text{CFP}} = \text{PIFS} + T_{\text{beacon}} + T_{\text{V-POLL}} + T_{\text{CF-End}} + 2 \times \text{SIFS}.$$

After broadcasting the beacon, the PC computes Δ_g, Δ_r,
and the variable $RACT := \Delta_r - (\delta_1 + \delta_2 + 3 \times \text{SIFS})$;
/∗ The variable $RACT$ denotes the remaining available channel
time if the PC proceeds to the next PE/PR or RE/RR handshake. ∗/
while ($\Delta_g > 0$ **and** $RACT > 0$
  and (registration process is not finished)) {
 The PC sends the PE/RE frame and announces $(\Delta_g, RACT)$;
 /∗ On receiving the PE/RE frame, each active real-time
  station, say A_k, takes the following admission test. ∗/
 if ($\mathcal{G}_{A_k} \leq \Delta_g$ **and** $\min\{\mathcal{G}_{A_k}, \mathcal{D}_{A_k}\} \leq RACT$)
  Station A_k replies the PR/RR frame and declares $(\mathcal{D}_{A_k}, \mathcal{G}_{A_k})$;
 $status$:= **receive** (PR or RR);
  /∗ The PC updates the channel state variable $status$
  according to received PR/RR frames. ∗/
 switch ($status$) {
  case SINGLE:
   The PC places the real-time station A_k on the polling list;
   $\Delta_g := \Delta_g - (\text{SIFS} + \mathcal{G}_{A_k} + \frac{6\times 8}{\text{CDR}})$;
   $\Delta_r := \Delta_r - (\delta_1 + \delta_2 + \min\{\mathcal{G}_{A_k}, \mathcal{D}_{A_k}\} + 3\text{SIFS} + \frac{6\times 8}{\text{CDR}})$;
   break;
  /∗ Note that the length of the V-POLL frame will increase
  by 6 bytes (48 bits) if a new real-time flow is admitted. ∗/
  case IDLE:
   $\Delta_r := \Delta_r - (\delta_1 + \text{PIFS})$; **break**;
  case COLLISION:
   $\Delta_r := \Delta_r - (\delta_1 + \delta_2 + 2 \times \text{SIFS})$; **break**;
 }
 $RACT := \Delta_r - (\delta_1 + \delta_2 + 3 \times \text{SIFS})$;
}

Fig. 7. The UPCF admission control algorithm.

- During the registration period, we let

$$\delta_1 = \begin{cases} T_{\text{PE}}, & \text{if the PC sends out the PE frame,} \\ T_{\text{RE}}, & \text{if the PC sends out the RE frame.} \end{cases}$$

$$\delta_2 = \begin{cases} T_{\text{PR}}, & \text{if the mobile replies the PR frame,} \\ T_{\text{RR}}, & \text{if the mobile replies the RR frame.} \end{cases}$$

- We define two auxiliary variables Δ_r and Δ_g respectively to assist the PC in verifying whether **P1** and **P2** are always satisfied, where

$$\Delta_r = \text{CFPMaxDuration} - \left[T_s + O_{\text{CFP}} + \sum_{A_i \in \mathcal{L}} \left(\text{SIFS} + \min\{\mathcal{D}_{A_i}, \mathcal{G}_{A_i}\}\right)\right] \quad (8)$$

and

$$\Delta_g = \text{CFPMaxDuration} - \left[\widehat{T}_s + O_{\text{CFP}} + \sum_{A_i \in \mathcal{L}} (\text{SIFS} + \mathcal{G}_{A_i})\right]. \quad (9)$$

Refer to Fig. 6, Fig. 7 specifies the admission control operations performed cooperatively by the PC and all active real-time stations during the registration period. Note that the RE frame format is shown in Fig. 5(d).

III. Throughput Analysis

We follow the analytic model proposed in [11] to evaluate the approximate throughput of the UPCF protocol in a WLAN which consists of one AP and N associated stations. We consider that there are merely real-time and non-real-time stations in a WLAN; that is, $H = 1$ and only the real-time stations can get on the polling list. Let N_{rt} and N_{nrt} be the number of real-time and non-real-time stations respectively, where $N_{rt} = 2^m$ and $N_{rt} + N_{nrt} = N$. For ease of analysis, we assume that, at the start of each CFP, each non-admitted real-time station has a probability p of intending to join the polling list. Let $R[i, N_{rt} - \ell]$ denote the probability that, given $|\mathcal{L}| = \ell$, there are i active real-time stations at the beginning of CFP, where $0 \leq i \leq N_{rt} - \ell$. Then we have

$$R[i, N_{rt} - \ell] = \binom{N_{rt} - \ell}{i} p^i (1-p)^{N_{rt} - \ell - i}.$$

In the following, we will derive the average length of the polling period, which starts from the V-POLL frame and finishes before the CF-End frame. Since N_{rt} is finite, given i active real-time stations, the deterministic tree-splitting algorithm ensures that the maximum length of the registration period is finite and its value only depends on N_{rt} and i. For simplicity, we consider a homogeneous CBR traffic scenario where $\mathcal{D} = \mathcal{D}_{A_i} = \mathcal{D}_{A_j}$, $\mathcal{G} = \mathcal{G}_{A_i} = \mathcal{G}_{A_j}$, and $\mathcal{D} = \mathcal{G} \leq T_{\max\text{MPDU}}$ for all real-time stations $A_i, A_j \in \mathcal{L}$ and $A_i \neq A_j$. As a result, the PC can allocate the same TXOP value, $\mathcal{D}$, to each polled station. By exploiting (9), the maximum number of admitted stations $\widehat{\ell}$ in UPCF is bounded by

$$\widehat{\ell} \leq \left\lfloor \frac{1}{\mathcal{G} + \text{SIFS}} (\text{CFPMaxDuration} - \widehat{T}_s - O_{\text{CFP}}) \right\rfloor. \quad (10)$$

Hence, we can tune the values of $\widehat{\ell}$ and SF such that the following inequality always holds:

$$\text{PIFS} + T_{\text{beacon}} + \text{SIFS} + \max\{T_{\text{reg}} + T_{\text{polling}}\} + T_{\text{CF-End}} \leq \text{SF} - (\text{CP}_{\min} + \widehat{T}_s)$$

where T_{reg} denotes the length of the registration period. Note that $\max\{T_{\text{reg}} + T_{\text{polling}}\}$ is finite and its value only relies on the number of active real-time stations, $\widehat{\ell}$, and N_{rt}. Under the circumstances, the PC is only required to control the polling list size. In other words, if there are already $\widehat{\ell}$ stations on the polling list, the registration process will be skipped until at least one admitted station get off the polling list. On the other hand, if there are only $\ell < \widehat{\ell}$ admitted stations, given i active real-time stations, the PC will keep executing the registration procedure until the $\min\{\widehat{\ell} - \ell, i\}$th SINGLE event occurs.

For each admitted real-time station, we assume that it will sojourn B superframes to complete its entire flow transmission, where B is a geometric random variable with parameter q, that is, $\Pr[B = b] = q(1-q)^{b-1}$ for $b = 1, 2, \ldots$. Because of the memoryless property of the geometric distribution, this assumption implies that each polled station will set more data bit to 0

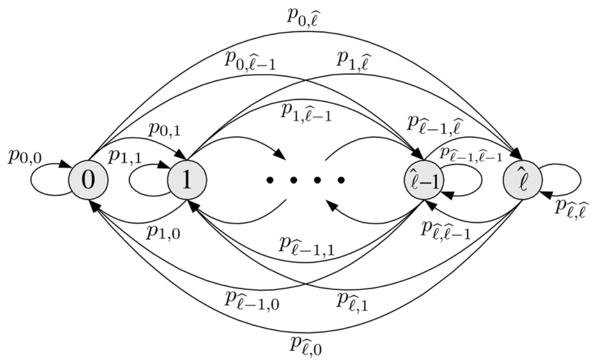


Fig. 8. Markov chain for the UPCF protocol.

with probability q when transmitting an MPDU. Given N_{rt} and ℓ, we define

$$W[i, N_{rt}-\ell] \equiv \begin{cases} R[i, N_{rt}-\ell], & \text{if } 0 \le i \le \widehat{\ell}-\ell-1, \\ \sum_{j=\widehat{\ell}-\ell}^{n} R[j, N_{rt}-\ell], & \text{if } \widehat{\ell}-\ell \le i \le N_{rt}-\ell. \end{cases}$$

The state transition diagram of the Markov chain is shown in Fig. 8, where the state π_ℓ represents the stationary probability that there are ℓ stations on the polling list. Note that the number of admitted stations can increase up to $\widehat{\ell}$ after the end of the registration period, but can decrease by up to ℓ after the end of the polling period. Let $p_{i,j}$ be the transition probability from state i to state j and X_t be the state index in the tth polling period. The transition probabilities can be specified as

$$p_{i,j} = \Pr[X_{t+1}=j \mid X_t = i] = \begin{cases} \sum_{k=0}^{j} \binom{i}{i-j+k} q^{i-j+k}(1-q)^{j-k} W[k, N_{rt}-i], \\ \quad \text{if } 0 \le j \le i \le \widehat{\ell}, \\ \sum_{k=0}^{i} \binom{i}{k} q^{k}(1-q)^{i-k} W[j-i+k, N_{rt}-i], \\ \quad \text{if } 0 \le i < j \le \widehat{\ell}. \end{cases}$$

Let $\boldsymbol{\pi} = [\pi_0, \pi_1, \ldots, \pi_{\widehat{\ell}}]$ be the stationary probability vector and $\boldsymbol{P} = [p_{i,j}]$ be the one-step transition probability matrix. The balance equation for this Markov chain is $\boldsymbol{\pi} = \boldsymbol{\pi P}$. From this, together with the normalization condition that $\sum_{\ell=0}^{\widehat{\ell}} \pi_\ell = 1$, we can obtain the vector $\boldsymbol{\pi}$. Hence, the average number of stations on the polling list is $\overline{\ell} = \sum_{\ell=0}^{\widehat{\ell}} \ell \times \pi_\ell$. Let $\overline{T}_{\text{polling}}$ denote the expected length of the polling period. We have the following result:

$$\overline{T}_{\text{polling}} = (1-\pi_0 \times R[0, N_{rt}])(T_{\text{V-POLL}}+\text{SIFS}) + \overline{\ell}(\mathcal{D}+\text{SIFS}). \quad (11)$$

Next, we would like to derive the average length of the prioritization period, which starts from the PE_1 frame and finishes before the RE frame. (Refer to Fig. 2). In the case when $0 \le \ell \le \widehat{\ell}-1$, the PC will perform the prioritization procedure. Thus, we have

$$\overline{T}_{\text{pri}} = (1-\pi_{\widehat{\ell}}) \times \left[R[0, N_{rt}-\ell] \times (T_{\text{PE}}+\text{PIFS}) + \sum_{i=1}^{N_{rt}-\ell} R[i, N_{rt}-\ell] \times (T_{\text{PE}}+T_{\text{PR}}+2\times\text{SIFS}) \right] \quad (12)$$

where $\overline{T}_{\text{pri}}$ denotes the expected length of the prioritization period and the fraction $1-\pi_{\widehat{\ell}}$ represents the fraction of non-skipped registration periods.

In what follows, we derive the average length of the collision resolution period, which starts from the RE frame and finishes before the V-POLL frame. Once the prioritization period ends in a COLLISION event, the PC then evenly splits the AddressPattern $\mathcal{A}$ involved in the collision along a dimension (bit) into two subsets, $\mathcal{A}_1$ and $\mathcal{A}_2$. Note that $|\mathcal{A}_1| = |\mathcal{A}_2| = |\mathcal{A}|/2$. The PC first recursively resolves the collision of $\mathcal{A}_1$, and then resolves the collision of $\mathcal{A}_2$ independently. Given $i \ge 2$ active real-time stations, the notations $\overline{C}(N_{rt}, i, k)$, $\overline{I}(N_{rt}, i, k)$, and $\overline{S}(N_{rt}, i, k)$ denote the average number of COLLISION steps (rounds), number of IDLE steps, and number of SINGLE steps, respectively, required to resolve collisions during the registration period until the kth SINGLE event takes place, where $2 \le k \le i \le N_{rt}$. Clearly, $\overline{S}(N_{rt}, i, k) = k$ regardless of N_{rt}. With each further splitting, the set of remaining possible numbers in one of the new subsets is halved. Besides, the number of active stations in the left or right subsets follows the hyper-geometrical distribution. Based on derivations in [11], $\overline{C}(N_{rt}, i, k)$ and $\overline{I}(N_{rt}, i, k)$ can be expressed in the following recursive forms:

$$\overline{C}(N_{rt}, i, k) = \begin{cases} 1 + \sum_{j=\nu}^{\omega} \frac{\binom{2^{m-1}}{i-j}\binom{2^{m-1}}{j}}{\binom{2^m}{i}} \Big[\overline{C}(2^{m-1}, i-j, i-j) \\ \quad + \overline{C}(2^{m-1}, j, k-i+j) \Big], \quad \text{if } k > i-j, \\ 1 + \sum_{j=\nu}^{\omega} \frac{\binom{2^{m-1}}{i-j}\binom{2^{m-1}}{j}}{\binom{2^m}{i}} \times \overline{C}(2^{m-1}, i-j, i-j), \\ \quad \text{otherwise} \end{cases} \quad (13)$$

where $\nu = \max\{0, i-2^{m-1}\}$, $\omega = \min\{0, 2^{m-1}\}$, and $N_{rt} = 2^m$. Besides,

$$\overline{I}(N_{rt}, i, k) = \begin{cases} \sum_{j=\nu}^{\omega} \frac{\binom{2^{m-1}}{i-j}\binom{2^{m-1}}{j}}{\binom{2^m}{i}} \Big[\overline{I}(2^{m-1}, i-j, i-j) \\ \quad + \overline{I}(2^{m-1}, j, k-i+j) \Big], \quad \text{if } k > i-j, \\ \sum_{j=\nu}^{\omega} \frac{\binom{2^{m-1}}{i-j}\binom{2^{m-1}}{j}}{\binom{2^m}{i}} \times \overline{I}(2^{m-1}, i-j, i-j), \\ \quad \text{otherwise.} \end{cases} \quad (14)$$

Notice that the actual values of $\overline{C}(N_{rt}, i, k)$ and $\overline{I}(N_{rt}, i, k)$ may be smaller than those calculated by (13) and (14) in that our tree-splitting algorithm can intelligently avoid some pointless polls. (See Fig. 3).

Let $\overline{T}_{\text{crp}}$ denote the expected length of the collision resolution period. Given ℓ already-admitted stations and $i \ge 2$ active real-time stations, $\overline{T}_{\text{crp}}$ equals the sum of the average number of SINGLE steps times their duration $T_{\text{RE}} + T_{\text{RR}} + 2 \times \text{SIFS}$, plus the average number of IDLE steps times their duration $T_{\text{RE}} + \text{PIFS}$, plus the average number of COLLISION steps times their duration $T_{\text{RE}} + T_{\text{RR}} + 2 \times \text{SIFS}$ until $\min\{i, \widehat{\ell}-\ell\}$ SINGLE events are recognized by the PC. Consequently, we have the following result:

$$\overline{T}_{\text{crp}} = (1-\pi_{\widehat{\ell}}) \times \left\{ \sum_{i=2}^{N_{rt}-\ell} R[i, N_{rt}-\ell] \times \left[(T_{\text{RE}}+\text{PIFS}) \times \overline{I}(N_{rt}, i, \widehat{\ell}-\ell) + (T_{\text{RE}}+T_{\text{RR}}+2\times\text{SIFS}) \times \left(\overline{S}(N_{rt}, i, \widehat{\ell}-\ell) + \overline{C}(N_{rt}, i, \widehat{\ell}-\ell) - 1 \right) \right] \right\}. \quad (15)$$

Note that, in (15), we subtract one from the value of $\overline{C}(N_{rt}, i, \widehat{\ell}-\ell)$ since it counts in one COLLISION step taking place in the prioritization period. Let $\overline{\text{CP}}$ be the expected length of the contention period. By exploiting (11), (12), and (15), we obtain

$$\overline{\text{CP}} = \text{SF} - (\text{PIFS} + T_{\text{beacon}} + \text{SIFS} + \overline{T}_{\text{pri}} + \overline{T}_{\text{crp}} + \overline{T}_{\text{polling}} + T_{\text{CF-End}}). \quad (16)$$

We are finally in the condition to determine the normalized throughput $\mathcal{S}$, defined as the fraction of time, during which the channel is being used to successfully transmit data frames. Let $\mathcal{S}_{\text{DCF}}(N_{nrt})$ be the normalized throughput of DCF in the presence of N_{nrt} non-real-time stations and its value can be found in [27]. When UPCF and DCF coexist in a WLAN, we can express $\mathcal{S}$ as the ratio

$$\mathcal{S} = \frac{E[\text{time used for successful data transmission in a superframe}]}{E[\text{length of a superframe}]} = \frac{\overline{\ell} \times \mathcal{D} + \overline{\text{CP}} \times \mathcal{S}_{\text{DCF}}(N_{nrt})}{\text{SF}}.$$

IV. Performance Evaluation

A. Simulation Model

We have developed event-driven simulators using Visual C to verify the performance of UPCF and compare our results to the PCF. The simulation model assumes there are one AP and 255 associated stations in a single WLAN cell. All simulation runs were carried out for a duration of 1.8×10^8 μs. Table II summarizes the system parameter values, which follow the IEEE 802.11 MAC specifications for the direct sequence spread spectrum (DSSS) physical layer [13]. Note that the power consumption parameters in Table II follow the specifications adopted in [15]. The following three types of traffic are considered in our simulations.

1) Data Traffic: Recently, extensive studies [2] have shown that the traditional Poisson process cannot capture any correlation between consecutive packet arrivals. Therefore, we adopt the two-state (ON/OFF) *Markov Modulated Poisson Process* (MMPP) data traffic model [2] to recover from this problem. The properties of a two-state MMPP process are as follows. 1) During the ON state, the arrival process is Poisson with rate λ. On the other hand, no traffic is generated during the OFF state. 2) The sojourn time in each state is exponentially distributed. The data payload size is fixed at 2304 bytes. Since the data frame is quite large, each DCF-station shall employe the RTS/CTS exchange procedure to transmit data frames.

TABLE II
SYSTEM PARAMETERS USED IN THE SIMULATION

Parameter	Value
Channel bit rate	11 Mbps
Superframe length	25 ms
SIFS	10 μs
SlotTime	20 μs
CWmin	31 slots
CWmax	1023 slots
RTS frame length	20 bytes
CTS/ACK frame length	14 bytes
Reassociation Request frame length	38 bytes
Reassociation Response frame length	34 bytes
Beacon frame length	57 bytes
PE frame length	17 bytes
CF-End frame length	20 bytes
Power consumption in TRANSMIT state	1.65 Watt
Power consumption in RECEIVE state	1.4 Watt
Power consumption in LISTEN state	1.15 Watt
Power consumption in DOZE state	0.045 Watt

TABLE III
TRAFFIC PARAMETER VALUES FOR DATA, CBR, AND VBR MODELS

Data Traffic Parameter	Value
Average ON duration	3.3 sec
Average OFF duration	22.8 sec
CBR Traffic Parameter	**Value**
Conversation length	180 sec
Principle talkspurt	1.00 sec
Principle silent gap	1.35 sec
Voice data bit rate	64 Kbps
Maximum tolerable frame delay ($D_{\max}^{\text{voice}}$)	25 ms
VBR Traffic Parameter	**Value**
Peak bit rate	420 Kbps
Minimum bit rate	120 Kbps
Mean bit rate	240 Kbps
Mean state holding time	160 ms
Mean video call length	180 sec
Maximum tolerable frame delay ($D_{\max}^{\text{video}}$)	50 ms

2) CBR Voice Traffic: The voice traffic is generally modelled as a two-state (ON/OFF) Markov process with talkspurt and silent states [23]. In the talkspurt state, the voice source generates a constant continuous bit-stream; in the silent state, no voice frame will be generated. Notice that, when measuring the UPCF capacity (maximum polling list size), we will consider the always-ON model; i.e., the silent duration is 0.

3) VBR Video Traffic: The video traffic can be mimicked by a multi-state model [1], [23] where a state generates a continuous bit-stream for a certain holding duration. The bit rate values of different states are obtained from a truncated exponential distribution with a minimum and a maximum bit rate values. The holding times of the states are assumed to be statistically independent and exponential distributed.

Table III summarizes the traffic parameter values. We assume that the voice traffic has the highest priority, the video traffic has

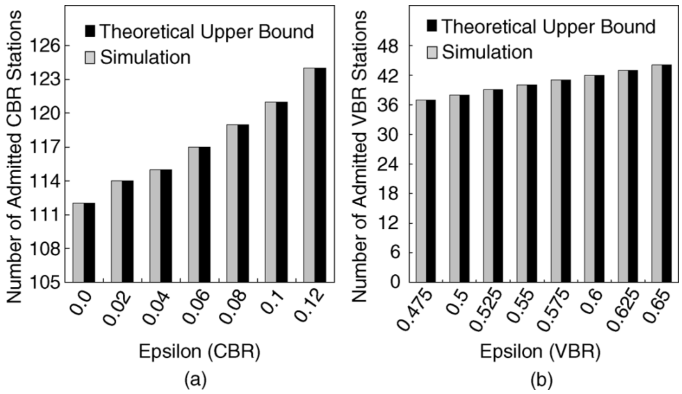


Fig. 9. The number of real-time stations admitted by the UPCF during the entire simulation time. (a) $N_{\rm CBR} = 150$, $N_{\rm VBR} = 0$, and $\lambda = 0.1$ frames/s/DCF-station. (b) $N_{\rm CBR} = 0$, $N_{\rm VBR} = 150$, and $\lambda = 0.1$ frames/s/DCF-station.

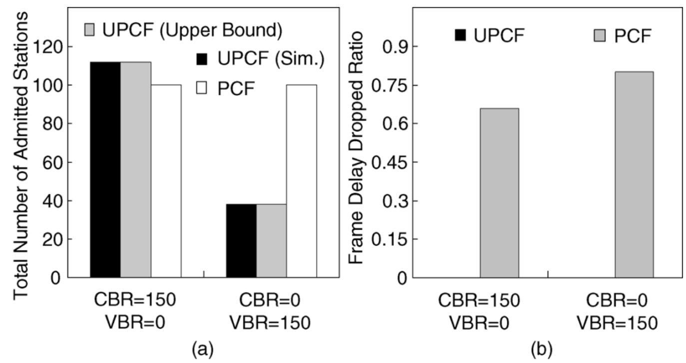


Fig. 10. (a) The number of real-time stations admitted by the UPCF/PCF during the entire simulation time. (b) Comparisons of the derived FDDR by UPCF/UPCF under the pure CBR/VBR traffic environments. ($\varepsilon_{\rm CBR} = 0.0$, $\varepsilon_{\rm VBR} = 0.5$, $N_{\rm DCF} = 105$, and $\lambda = 0.5$ frames/s/DCF-station.).

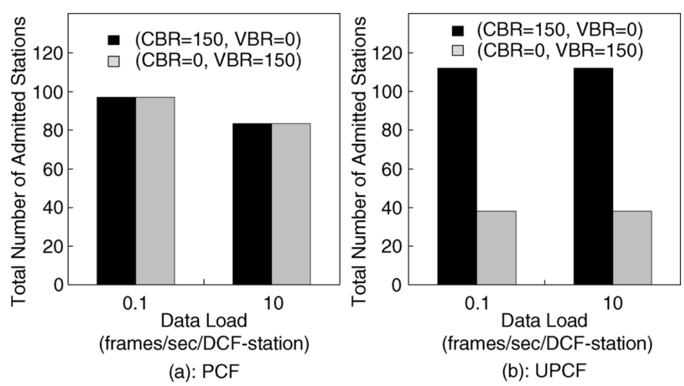


Fig. 11. The number of CBR/VBR stations admitted by the UPCF/PCF during the entire simulation time under the different DCF traffic load. ($N_{\rm DCF} = 105$, $\varepsilon_{\rm CBR} = 0.0$, and $\varepsilon_{\rm VBR} = 0.5$.).

the second highest priority, and the data traffic has the lowest priority. Voice and video frames that cannot be transmitted within their respective maximum tolerable time ($D_{\max}^{\rm voice}$ and $D_{\max}^{\rm video}$) will be dropped. Hence, we have ${\rm SF} \le \min\{D_{\max}^{\rm voice}, D_{\max}^{\rm video}\} =$ 25 ms. Let $N_{\rm DCF}$, $N_{\rm PS}$, $N_{\rm CBR}$, and $N_{\rm VBR}$ be the number of DCF, PS, CBR, and VBR associated stations, respectively. We fix $N_{\rm DCF}+N_{\rm PS}+N_{\rm CBR}+N_{\rm VBR} = 255$. For ease of exposition, each CBR (VBR, respectively) station selects the same value of $\varepsilon_{\rm CBR}$ ($\varepsilon_{\rm VBR}$, respectively) and declares the same value of $\mathcal{G}_{\rm CBR}$ ($\mathcal{G}_{\rm VBR}$, respectively). Besides, we assume that: 1) all associated stations are pollable; 2) DCF stations are always awake; 3) PS stations cannot generate any traffic; and 4) PCF uses the round-robin scheme to schedule the polling order. Note that a pollable station may not be on the polling list. Unlike the immediate-ACK policy used in PCF, we adopt the no-ACK policy in UPCF to transport the CBR/VBR frames. Normally, each CBR/VBR station selects one destination randomly and uniformly from the AP and remaining 244 stations. However, when examining the power saving capacity, it intentionally selects a different PS station as its destination. In our experiments, we did not count the energy consumption of the AP since it is often considered to have unlimited power resources.

B. Simulation Results

To verify the accuracy of the run-time admission control algorithm, we measure the UPCF capacity (the maximum number of real-time stations that the PC can admit) under the pure CBR/VBR traffic environments through the simulation program and (10). For the pure CBR traffic scenario, we use $N_{\rm CBR} = 150$ and $N_{\rm VBR} = 0$; for the pure VBR traffic scenario, we use $N_{\rm VBR} = 150$ and $N_{\rm CBR} = 0$. Fig. 9 shows that, no matter how the value of $\varepsilon_{\rm CBR}$ or $\varepsilon_{\rm VBR}$ varies, the maximum polling list size in UPCF exactly matches the IEEE 802.11 theoretical upper bound. These results justify the superiority of the mobile-assisted admission control scheme.

Next, we investigate the relationship between the maximum polling list size and the *frame delay dropped ratio* (FDDR). The FDDR is defined as the fraction of dropped voice/video frames caused by violating the delay constraints. Recall that, in IEEE 802.11, data transfer between stations in an infrastructure WLAN should be relayed through the AP. And we observe that almost all traffic is station-to-station. This implies that, in the saturated condition, the optimal polling list size in PCF should be about *half* that of the UPCF. Since PCF does not perform any admission control, the PC can admit a large number of real-time stations, which may be far beyond its capacity. In this case, several real-time stations will not be polled during the entire CFP. As a result, the FDDR of PCF shown in Fig. 10(b) is remarkably large. On the other hand, Fig. 10(b) shows a surprising result that the FDDR of UPCF is very close to 0 even $\varepsilon_{\rm VBR} = 0.5$. The reasons are as follows. Let $\mathcal{D}_{\rm VBR}$ be a random variable equal to the demanded TXOP of each VBR station. By definition, when $\varepsilon_{\rm VBR} = 0.5$, $\mathcal{G}_{\rm VBR}$ is the *median* of $\mathcal{D}_{\rm VBR}$ [20]; that is, $\Pr[\mathcal{D}_{\rm VBR} \ge \mathcal{G}_{\rm VBR}] = \Pr[\mathcal{D}_{\rm VBR} \le \mathcal{G}_{\rm VBR}]$. Recall that dynamic TXOP allocation scheme tries to allocate unused channel time from those VBR stations whose TXOP needs are less than $\mathcal{G}_{\rm VBR}$ to those stations whose TXOP needs are greater than $\mathcal{G}_{\rm VBR}$. Since the cardinalities of these two sets are statistically equal, we can expect that, in this case, the FDDR is very close to 0.

To acquire contention-free services, we hope that real-time stations can promptly register with the PC and the registration process should not be adversely affected by the low-priority (DCF) stations. Since, in IEEE 802.11, the registration (reassociation) process relies on DCF, we can find that, in Fig. 11(a), the number of admitted real-time stations decreases as the data load becomes heavier. However, in Fig. 11(b), we observe that

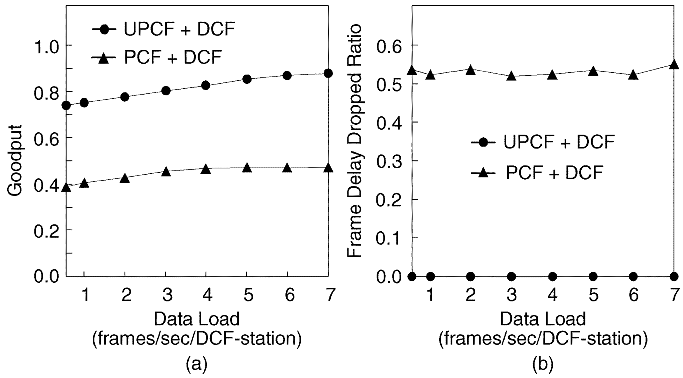


Fig. 12. Performance comparison for (UPCF+DCF) versus (PCF+DCF) under the heterogeneous traffic scenarios. (a) Goodput. (b) FDDR. ($N_{\rm PS} = 0$, $\varepsilon_{\rm CBR} = 0.0$, and $\varepsilon_{\rm VBR} = 0.5$.).

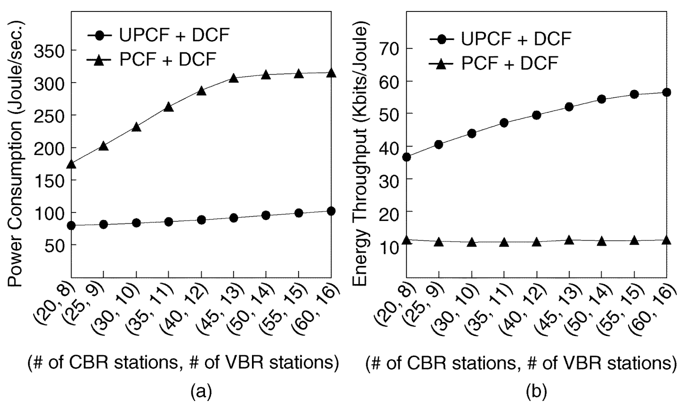


Fig. 13. Performance comparison for (UPCF+DCF) versus (PCF+DCF) under the heterogeneous traffic scenarios. (a) Power consumption. (b) Energy throughput. ($\varepsilon_{\rm CBR} = 0.0$, $\varepsilon_{\rm VBR} = 0.5$, $N_{\rm DCF} = 55$, and $\lambda = 3.0$ frames/s/DCF-station.).

the number of admitted real-time stations in UPCF is not affected by the behaviors of non-real-time stations at all. This result justifies the creation of the registration period dedicated to real-time stations.

This time we examine the goodput and FDDR of UPCF and PCF under the heterogeneous traffic scenarios. The *goodput* (normalized effective throughput) is defined as the fraction of time devoted by real-time (CBR/VBR) stations and non-real-time (DCF) stations to successfully send their pure payload to their respective destinations. We use $N_{\rm CBR} = 100$, $N_{\rm VBR} = 30$, $N_{\rm PS} = 0$, $\varepsilon_{\rm CBR} = 0.0$, and $\varepsilon_{\rm VBR} = 0.5$. For CBR sources, we consider the ON-OFF model. Fig. 12(a) reveals that the peak goodput of UPCF achieves nearly 90%, while the peak goodput of PCF is only around 47%. Fig. 12(b) reveals that the FDDR of UPCF is very close to 0 regardless of DCF traffic load; by contrast, the FDDR of PCF is over 50% in most situations.

Then we explore the power management issues of UPCF and PCF under the heterogeneous traffic scenarios. We fix $N_{\rm DCF}$ as 55 and vary $(N_{\rm CBR}, N_{\rm VBR})$ from (20, 8) to (60, 16) to observe its effect on the power consumption and energy throughput. The *energy throughput* is defined as the amount of successful data delivered per Joule of energy. It is obtained by dividing the total number of payload bits successfully sent from sources to destinations by the total energy consumption of all stations during the entire simulation time. Ref. [15] pointed out that energy throughput is also an important gauge since some MAC protocol may consume very little energy, but also contribute very little throughput. From Fig. 13(a), we find that UPCF is only about one-half or one-third the power consumption of PCF. The reasons are mainly twofold. 1) In the merger of UPCF and DCF, the power consumption of DCF stations $\mathcal{P}_{\rm DCF}$ is nearly unchanged regardless of $(N_{\rm CBR}, N_{\rm VBR})$. However, in 802.11, as the DCF load becomes severer due to the increase of both $N_{\rm CBR}$ and $N_{\rm VBR}$, $\mathcal{P}_{\rm DCF}$ keeps rising and is finally saturated at $(N_{\rm CBR} = 45, N_{\rm VBR} = 13)$. 2) Both the power consumptions of CBR/VBR/PS stations in UPCF and PCF respectively grow virtually with the increasing number of $N_{\rm CBR}$ and $N_{\rm VBR}$. Unfortunately, in PCF, all admitted stations are not allowed to sleep during the entire CFP [13]. Worse, PS stations whose corresponding TIM bits are set in the beacon frame will stay active during the entire CFP since they do not know when the PC will deliver the buffered data (of a flow) for them. Worst, active real-time stations that cannot reassociate with the PC successfully will remain awake during the entire CFP since they do not know when the next CP will arrive. As to the UPCF, after inspecting the V-POLL frame, any stations that can be neither a transmitter nor a receiver during the polling period can immediately return to the doze state. Further, UPCF employs the power-conserving scheduling (see Section II-E) such that stations that need to take part in the polling activity can spend as little awake time as possible. Fig. 13(b) shows that as both $N_{\rm CBR}$ and $N_{\rm VBR}$ increase, the energy throughput of UPCF increases rapidly, while the energy throughput of PCF is almost around 11 Kbits/Joule.

V. Conclusion

It is evident that a wireless MAC protocol optimized for *joint* quality-of-service and power conservation can let mobile users enjoy multimedia applications over a long period of time. For WLANs, there has been extensive work in the literature on improving IEEE 802.11 standard by providing better power management mechanisms [15], [26] as well as by supporting beyond best-effort services [4], [8], [9], [11], [14], [17], [21]–[23], [25]. However, they present separated efforts in both domains and their solutions may not combine well with each other. From Table I, we can find what makes our proposed solution, Unified Point Coordination Function (UPCF), unique is that it integrates non-reversal prioritization, time-bounded reservation, peer-to-peer communication support, dynamic channel time allocation, probabilistic per-flow bandwidth assurance, mobile-assisted admission control, and optimized power management into *one* simple scheme to handle different types of real-time multimedia traffic. To our best knowledge, none of the proposed 802.11-compliant MAC protocols provides all such characteristics. The performance of UPCF has been evaluated via both mathematical analysis and simulation experiments. Simulation results demonstrate that, in comparison with PCF, UPCF not only provides higher goodput and energy throughput, but also achieves lower power consumption and frame delay dropped ratio. Above all, due to the independence of the channel access mechanisms operating in the CP, UPCF can pass the current Wi-Fi certification and may coexist with the forthcoming IEEE 802.11e standard [14].

REFERENCES

[1] I. F. Akyildiz, D. A. Levine, and I. Joe, "A slotted CDMA protocol with BER scheduling for wireless multimedia networks," *IEEE/ACM Trans. Netw.*, vol. 7, no. 2, pp. 146–158, Apr. 1999.

[2] G. Anastasi and L. Lenzini, "QoS provided by the IEEE 802.11 wireless LAN to advanced data applications: A simulation analysis," *ACM/Kluwer Wireless Netw.*, vol. 6, pp. 99–108, 2000.

[3] G. Anastasi, L. Lenzini, and E. Mingozzi, "HIPERLAN/1 MAC protocol: Stability and performance analysis," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 9, pp. 1787–1798, Sep. 2000.

[4] M. Barry, A. T. Campbell, and A. Veres, "Distributed control algorithms for service differentiation in wireless packet networks," in *Proc. IEEE INFOCOM*, Apr. 2001, pp. 582–590.

[5] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1992.

[6] S. Choi, "Emerging IEEE 802.11e WLAN for quality-of-service (QoS) provisioning," *SK Telecom Telecommun. Rev.*, vol. 12, no. 6, pp. 894–906, Dec. 2002.

[7] Z.-T. Chou, "Hybrid multi-access protocols for real-time transmission and power conservation in wireless networks," Ph.D. dissertation, National Taiwan Univ., Taiwan, R.O.C., 2003.

[8] D.-J. Deng and R.-S. Chang, "A priority scheme for IEEE 802.11 DCF access method," *IEICE Trans. Commun.*, pp. 96–102, Jan. 1999.

[9] M. Fischer, "QoS baseline proposal for the IEEE 802.11e," IEEE Doc. 802.11-00/360.

[10] A. Ganz, A. Phonphoem, and Z. Ganz, "Robust SuperPoll with chaining protocol for IEEE 802.11 wireless LANs in support of multimedia applications," *ACM/Kluwer Wireless Netw.*, vol. 7, pp. 65–73, 2001.

[11] R. Garcés, "Collision avoidance and resolution multiple access," Ph.D. dissertation, Univ. California, Santa Cruz, CA, 1999.

[12] HelloWLAN. [Online]. Available: http://www.hellosoft.com/products/hwlan/hwlan11mac.htm

[13] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE 802.11 Standard, 1999.

[14] *Draft Amendment to Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)*, IEEE 802.11e/D6.0, 2003.

[15] E.-S. Jung and N. H. Vaidya, "An energy efficient MAC protocol for wireless LANs," in *Proc. IEEE INFOCOM*, Jun. 2002, vol. 3, pp. 1756–1764.

[16] S. Keshav, *An Engineering Approach to Computer Networking: ATM Networks, the Internet, and the Telephone Network*. Reading, MA: Addison-Wesley, 1997.

[17] S.-C. Lo, G. Lee, and W.-T. Chen, "An efficient multipolling mechanism for IEEE 802.11 wireless LANs," *IEEE Trans. Comput.*, vol. 52, no. 6, pp. 764–778, Jun. 2003.

[18] K. Nakano and S. Olariu, "Randomized initialization protocols for ad hoc networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 11, no. 7, pp. 749–759, Jul. 2000.

[19] W. Pattara-Atikom, P. Krishnamurthy, and S. Banerjee, "Distributed mechanisms for quality of service in wireless LANs," *IEEE Wireless Commun.*, pp. 26–34, Jun. 2003.

[20] S. Ross, *A First Course in Probability*, 5th ed. Englewood Cliffs, NJ: Prentice-Hall, 1998.

[21] O. Sharon and E. Altman, "An efficient polling MAC for wireless LANs," *IEEE/ACM Trans. Netw.*, vol. 9, no. 4, pp. 439–451, Aug. 2001.

[22] J.-P. Sheu, C.-H. Liu, S.-L. Wu, and Y.-C. Tseng, "A priority MAC protocol to support real-time multimedia traffic in ad hoc networks," *ACM/Kluwer Wireless Netw.*, vol. 10, pp. 61–69, Jan. 2004.

[23] S.-T. Sheu and T.-F. Sheu, "A bandwidth allocation/sharing/extension protocol for multimedia over IEEE 802.11 ad hoc wireless LANs," *IEEE J. Sel. Areas Commun.*, vol. 19, pp. 2065–2080, Oct. 2001.

[24] A. Silberschatz and P. B. Galvin, *Operating System Concepts*, 4th ed. Reading, MA: Addison-Wesley, 1994.

[25] J. L. Sobrinho and A. S. Krishnakumar, "Quality-of-service in ad hoc carrier sense multiple access wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 17, pp. 1353–1368, Aug. 1999.

[26] J. A. Stine and G. D. Veciana, "Improving energy efficiency of centrally controlled wireless data networks," *ACM/Kluwer Wireless Netw.*, vol. 8, pp. 681–700, Nov. 2002.

[27] Y. C. Tay and K. C. Chua, "A capacity analysis for the IEEE 802.11 MAC protocol," *ACM Wireless Netw.*, vol. 7, pp. 159–171, 2001.

[28] VT6655 WLAN Controller. [Online]. Available: http://www.vntek.com/en/products/wireless/vt6655

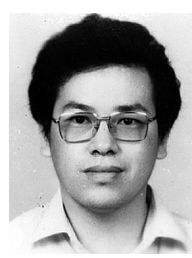
[29] Wi-Fi System Interoperability Test Plan. Dec. 2001 [Online]. Available: http://www.vocollect.com/us/products/11b_test_matrix.pdf

[30] WL6000 MAC Controller. [Online]. Available: http://www.duolog.com/wlanmac.html

Zi-Tsan Chou (S'98–M'03) received the B.S. degree in applied mathematics from National Chengchi University, the M.B.A. degree in information management from National Taiwan University of Science and Technology, and the Ph.D. degree in computer science and information engineering from National Taiwan University, Taipei, Taiwan, R.O.C., in 2003.

He is currently a Senior Engineer with the Networks and Multimedia Institute at the Institute for Information Industry, Taipei. His research interests include wireless medium access control, mobility management, and multimedia applications.

Dr. Chou received the Innovative and Prospective Technologies Project Award and the Creative Technology Application Award from the Institute for Information Industry in 2003 and 2004, respectively.



Ching-Chi Hsu received the B.S. degree in physics from National Tsing Hua University, Hsinchu, Taiwan, R.O.C., in 1971, and the M.S. and Ph.D. degrees in computer engineering from the Department of Electric Engineering, National Taiwan University, Taipei, Taiwan, in 1975 and 1982, respectively.

In 1977, he joined the faculty of the Department of Computer Science and Information Engineering at National Taiwan University and became an Associate Professor in 1982. Between 1987 and 2002, he was first engaged as a Professor and became the Chairman of the department. After serving in National Taiwan University for over 25 years, he became the President of Kai Nan University in 2002. Beginning in February 2004, he has been the Executive Vice President of the Institute for Information Industry, Taiwan, in which he is mainly in charge of accelerating the growth of information industry in the whole nation. His research interests include distributed processing of data and knowledge, mobile computing and wireless networks.

Shin-Neng Hsu received the M.S. degree in computer science and information engineering from National Taiwan University, Taipei, Taiwan, R.O.C., in 2003.

Currently, he is a Senior Engineer in the Information Technology Department of Taiwan Semiconductor Manufacturing Company, where his research focuses on wireless LANs and personal communication services.