

A power-saving scheduling for infrastructure-mode 802.11 wireless LANs [☆]

Hsiao-Po Lin, Shih-Chang Huang, Rong-Hong Jan ^{*}

Department of Computer and Information Science, National Chiao Tung University, Hsinchu 30050, Taiwan

Available online 20 March 2006

Abstract

This paper presents a novel method to arrange wakeup schedule for sleeping stations such that the number of wakeup stations in each beacon interval is balanced in IEEE 802.11 wireless local area networks (WLANs). This method reduces the probability of collision and thus the station can save more power. Next, we consider how to poll the wakeup stations to send the PS-Poll frame to get back their buffered data so that the contention can be avoided. Three different access scheduling mechanisms are proposed for the contention avoidance. In the first mechanism, only one of wakeup stations is scheduled to access the buffered data. The second and third mechanisms based on the smallest association ID (AID) first and the smallest queue length first, respectively, arrange a subset of wakeup stations to get back their buffered data within a beacon interval. Simulation results show that the proposed methods are effective in the power-saving.

© 2006 Elsevier B.V. All rights reserved.

Keywords: Wireless LAN; Infrastructure mode; Power saving; Traffic scheduling

1. Introduction

Recently, due to the technology explosion in wireless communication (e.g., Bluetooth, IEEE 802.11, GSM/GPRS, and WCDMA) and portable communication devices (e.g., notebook PCs, personal digital assistants, and smart phones), it has become possible for people to connect to the Internet anytime and anywhere, and remain on-line while roaming. Among these wireless communication techniques, IEEE 802.11 wireless local area networks (WLANs) [1–3] are the most widely-used local wireless network system in schools, offices, airports, etc. Besides, almost portable devices can be equipped to access IEEE 802.11 WLANs. However, the energy sources of these

portable devices come from their equipped batteries. Once the battery has run down, it needs to be recharged and portable device suffers a loss of network connectivity. Thus, the power-saving problem becomes an important issue for prolonging the operation of a portable device [4].

For IEEE 802.11 WLANs, a common method for power-saving is to powering down the transceiver. When transceiver is off, we say the mobile station is in *sleeping*. A listen-interval is a period of time for which the mobile station may choose to sleep. Power conservation in IEEE 802.11 can be achieved by maximizing the listen-interval. However, longer listen-intervals increase the transmission delay. In [5] and [6], they proposed listen-interval adaptation mechanisms for power-saving in which the mobile station dynamically adapts the duration of listen-interval according to the traffic situation. Thus, if the traffic load is light, the mobile device can set a longer listen-interval to save more power. Another kind of listen interval controlling is the quorum based scheme [7,8]. In the quorum based scheme, each station in power-saving mode synchronizes its wakeup schedule with each other such that the

[☆] This research was supported in part by the National Science Council, Taiwan, under Grant NSC 94-2219-E-009-005 and NSC94-2752-E-009-005-PAE, in part by the communication software technology of III, Taiwan, and in part by the Intel.

^{*} Corresponding author. Fax: +886 3 5721490.

E-mail address: rhjan@cis.nctu.edu.tw (R.-H. Jan).

station can deliver buffered frames to a power-saving station at right time when the radio of power-saving station is turned on.

Controlling the transmission power is another way to save mobile devices power [6]. The basic idea is using the least power to transmit data. In [6], an adaptive transmission power mechanism is proposed in which the access point (AP) computes the optimal transmission power for each associated mobile station based on the received signal and informs the mobile devices their optimal transmission power. Then, the mobile station can adjust its transmission power to the optimal value to saving power.

Frame aggregation can also be used to save power. There are two kinds of frame aggregation. One is combining multiple relative frames into an aggregated frame [9]. The other is compressing the payload to reduce the amount of transmission data.

Another interesting way to saving power is using a low power radio module, such as Bluetooth [10], or a special signaling channel [11], to actuate the normal IEEE 802.11 radio circuit. In [10], the Bluetooth module is used for signaling and IEEE 802.11 for data transmission. This method indeed can save mobile station power but the cost is that it needs an extra hardware. The method in [11] is similar to that in [10] except the low power radio module.

TDMA-based (time-division multiple-access) approaches [12,13] can also assist mobile stations to conserve power by scheduling channel access in advance so that mobile stations can turn their transceivers off when it is not their turn to transmit or receive. In [12], the access point periodically broadcasts a schedule frame containing start time and duration of time slot for each mobile station. The mobile station may send or receive frames only during its time slot and outside its time slots, the mobile station can turn its transceiver off to save power. In [13], they only focus on the point coordination function (PCF). When the PCF is used, time on the medium is divided into the contention free period (CFP) and the contention period (CP). In a CP, the point coordinator learns what traffic needs to be transmitted and then directs its transmissions in a CFP. In [14], they propose a power-saving algorithm that incorporates a contention-free scheduling function for data transmission in 802.11 ad hoc networks.

This paper focuses on the power management problem for an infrastructure mode IEEE 802.11 WLAN. We present a novel method to arrange wakeup schedule for sleeping stations such that the number of wakeup stations in each beacon interval is balanced in IEEE 802.11 WLANs. This method can reduce the probability of collision and thus the station saves more power. Next, we consider how to poll the wakeup stations to send the PS-Poll frame to get back their buffered data so that the contention can be avoided. Three different access scheduling mechanisms are proposed for the contention avoidance. In the first mechanism, only one of wakeup stations is scheduled to access the buffered data. The second and third mechanisms schedule a subset of wakeup stations to retrieve their buffered data within a beacon interval. The access sequences of the second and third mechanisms are based on the smallest association ID (AID) first and the smallest queue length first, respectively.

The rest of this paper is organized as follows. Section 2 reviews the operation of power saving mode in IEEE 802.11 WLANs. A wakeup scheduling problem is considered in Section 3 and three contention avoidance mechanisms for polling wakeup stations are presented in section 4. In Section 5, we give the simulation results to show the effectiveness of our proposed methods. Finally, the conclusion and possible future research are given in Section 6.

2. Power management in 802.11 WLANs

In infrastructure mode IEEE 802.11 WLANs, a mobile station can power down its transceiver and enter the power-saving mode (PS mode) for conserving power. The station can communicate its power management state to its AP. Thus, an AP knows the power management state of every station that has associated with it. When a frame arrives, the AP can determine whether the frame should be delivered to wireless network because the station is awake or buffered because the station is in PS mode.

After buffering frames, the next job for AP is to announce periodically which stations have frames waiting for them. That is, AP periodically broadcasts beacon frames with a traffic indication map (TIM) to its service stations. The TIM is a virtual bitmap in which each bit

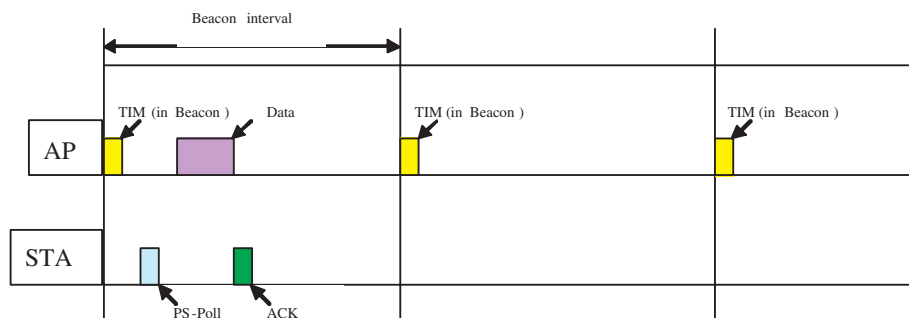


Fig. 1. IEEE 802.11 power saving mode.

Table 1
A sequence of $w_i(t)$ and $n(t)$

t	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$w_A(t)$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
$w_B(t)$	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
$w_C(t)$	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0
$w_D(t)$	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
$w_E(t)$	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
$w_F(t)$	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1
$n(t)$	3	2	1	3	2	3	3	2	1	3	2	3	3	2	1	3	2	3

Table 2
A sequence of $w_i(t)$ and $n(t)$ after station J joins in power saving mode

t	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$w_A(t)$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
$w_B(t)$	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
$w_C(t)$	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0
$w_D(t)$	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
$w_E(t)$	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
$w_F(t)$	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1
$w_J(t)$	–	–	–	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1
$n(t)$	3	2	1	3	2	4	3	2	2	3	2	4	3	2	2	3	2	4

Table 3
A sequence of $w_i(t)$ and $n(t)$ for station J with first wakeup time $t = 5$

t	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$w_A(t)$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
$w_B(t)$	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
$w_C(t)$	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0
$w_D(t)$	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
$w_E(t)$	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
$w_F(t)$	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1
$w_J(t)$	–	–	–	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
$n(t)$	3	2	1	3	3	3	3	3	1	3	3	3	3	3	1	3	3	3

corresponds to a particular AID. When a station has associated to an AP, it receives an AID from the AP. The AP sets the bit in TIM if it has buffered frames for the station with AID corresponding to the bit position.

Mobile stations in power saving mode have to wake up to listen for beacon frames and check the TIM. By this way, a mobile station can determine whether the AP has buffered frames for it. If the AP seldom buffers frames for the station, the station does not require waking up to check every beacon frame. Instead, it wakes up every *listen-interval* to check the beacon frame. A listen-interval is a number of beacon interval for which the mobile station may choose to sleep. If the station finds that the AP has buffered data for it, it will send a PS-Poll control frame to retrieve the buffered frames. When multiple stations have buffered frames, all stations with buffered frame contend the medium for sending PS-Poll. After sending the PS-Poll, a station has to awake until the buffered frames are received or the bit in the TIM corresponding its AID is no longer set.

For example, as shown in Fig. 1, a station, denoted as STA, is wakeup in the first beacon interval and receives the beacon frame in which the TIM indicates buffered data for it. Then, STA contends the medium for sending a PS-Poll frame to inform the AP that it is wakeup and ready to get back the buffered data. After AP receives the PS-Poll, it transmits a buffered frame to the STA. The STA returns an ACK frame to inform AP that the frame is received completely.

3. Load-aware wakeup scheduling

Consider a wireless LAN having an AP and six sleeping stations, $A, B, C, D, E,$ and F . The listen-intervals of stations, $A, B, C, D, E,$ and F are 1, 2, 3, 6, 6, and 6, respectively. Let $w_i(t)$ be an indication bit where $w_i(t) = 1$ if the station i wakes up at beacon interval t ; $w_i(t) = 0$, otherwise. Let $n(t)$ denote the total number of stations waking up at beacon interval t . Then, $n(t)$ can be found by

$$n(t) = \sum_{i \in S} w_i(t)$$

where set S includes all sleeping stations. Table 1 shows a sequence of $w_i(t)$ for each station and total number of wakeup stations $n(t)$, $t = 1, 2, \dots, 18$.

Assume that a station J enters the power saving mode at beacon interval 3 and its listen-interval is 3. Table 2 shows a sequence of $w_i(t)$ and $n(t)$, $t = 1, 2, \dots, 18$ after station J joins in power saving mode. Note that there are 4 stations that will wake up at beacon intervals 6, 12, and 18. If these stations (i.e., stations, A, B, F, and J) find that the AP has buffered data for them at beacon intervals 6, 12, or 18, they may suffer the collisions for sending PS-Poll to retrieve the buffered data. Note that collision causes retransmission and thus the station consumes more power for retrieving the buffered data. If we can arrange the first wakeup time for station J , the maximum of $n(t)$ can be reduced. This can save mobile stations' power. For example, if we can schedule the first wakeup time for station J at beacon interval 5, the maximum number of contending stations will be reduced to 3 (see Table 3).

In order to trace the wakeup time of each station, AP needs to maintain a wakeup counter, denoted as $c_i(t)$, for each sleeping station i . The $c_i(t)$ indicates remaining beacon intervals that station i will wake up. Initially, AP sets $c_i(t) = \ell_i - 1$ for station i whenever station i enters the sleeping mode where ℓ_i is the listen-interval of station i . The counter $c_i(t)$ will be decreased by one after beacon frame is transmitted. A station i wakes up if its $c_i(t)$

becomes zero. After that, the counter will reset to $c_i(t) = \ell_i - 1$ for further counting down. Thus, after transmitting a beacon frame, AP sets counter $c_i(t + 1), i \in S$, for beacon interval $t + 1$ as follows:

$$c_i(t + 1) = \begin{cases} c_i(t) - 1, & \text{if } c_i(t) \neq 0 \\ \ell_i - 1, & \text{if } c_i(t) = 0 \end{cases}$$

In general, if a mobile station i has wakeup counter $c_i(t)$ and listen-interval ℓ_i at beacon interval t , AP can find $w_i(t + k)$ for beacon interval $t + k$ by

$$w_i(t + k) = \begin{cases} 1 & \text{if } k \bmod \ell_i = c_i(t) \\ 0 & \text{otherwise} \end{cases}$$

where $k = 1, 2, \dots$. Then, $n(t + k)$ can be found by computing $n(t + k) = \sum_{i \in S} w_i(t + k)$ for $k = 1, 2, \dots$ where S includes all sleeping stations.

The wakeup scheduling problem (WSP) considered in this paper can be stated formally as follows: given a set of sleeping station S at beacon interval t consisting m stations and each of the station i having wakeup counter $c_i(t)$ and ℓ_i , for a new sleeping station j assign an initial value to its $c_j(t)$ such that the maximum value of $n(t + k) = \sum_{i \in S \cup \{j\}} w_i(t + k), k = 1, 2, \dots$ is minimized.

By observing the sequence of $n(t)$ in Table 1, we find that a pattern repeats every six beacon intervals, e.g., $(n(1), n(2), \dots, n(6)) = (n(7), n(8), \dots, n(12)) = (n(13), n(14), \dots, n(18)) = (3, 2, 1, 3, 2, 3)$. The length of this repeating pattern, r , can be found by computing the least common multiple (lcm) of listen-interval $\ell_i, i \in S$. For example, the

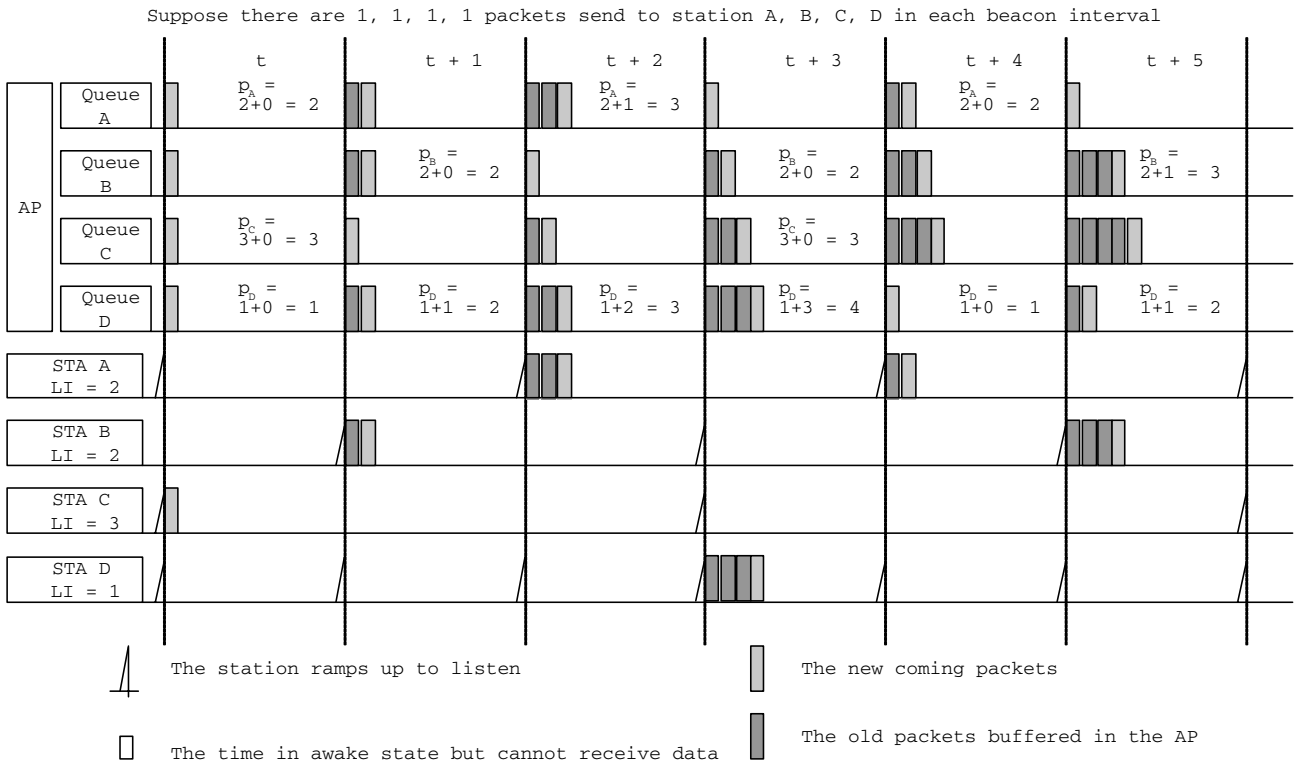


Fig. 2. Multiple wakeup single access.

listen-intervals of stations, *A, B, C, D, E,* and *F* are 1, 2, 3, 6, 6, and 6, respectively, in Table 1. Then

$$r = lcm\{1, 2, 3, 6, 6, 6\} = 6$$

Thus, $n^* = \max\{n(t+1), n(t+2), \dots, n(t+r)\} = \max\{n(t+k) | k = 1, 2, \dots\}$. Now, we want to add a new sleeping station *j* with listen-interval ℓ_j to the sleeping set *S* with repeating pattern size *r* and assign an initial value to $c_j(t)$. A stepwise solving method for WSP problem is given as follows.

1. Find $r = lcm\{\ell_j, r\}$ and a sequence of total number of wakeup stations $(n(t+1), n(t+2), \dots, n(t+r))$ for the first *r* intervals for sleeping station set $S \cup \{j\}$.
2. For $i = \ell_j - 1, \dots, 1, 0$, perform the following operations:
 - (a) Set $c_j(t) = i$ and find $(w_j(t+1), w_j(t+2), \dots, w_j(t+r))$;
 - (b) Set $(n(t+1), n(t+2), \dots, n(t+r)) = (n(t+1), n(t+2), \dots, n(t+r)) + (w_j(t+1), w_j(t+2), \dots, w_j(t+r))$;
 - (c) Find $n_i = \max\{n(t+1), n(t+2), \dots, n(t+r)\}$.
3. Find $n^* = \min\{n_i | i = \ell_j - 1, \ell_j - 2, \dots, 0\}$, say $n^* = n_k$, and thus set $c_j(t) = k$.

For example, six stations with $r = 6$ as given in Table 1, station *J* with $\ell_j = 3$ enters the sleeping mode. The AP applies the above solving method to determine the initial value of counter $c_j(t)$ for station *J* as follows.

1. $r = lcm\{3, 6\} = 6$ and $(n(t+1), n(t+2), \dots, n(t+6)) = (3, 2, 1, 3, 2, 3)$
2. $i = 2$:

- (a) Set $c_j(t) = 2$ and find $(w_j(t+1), w_j(t+2), \dots, w_j(t+6)) = (0, 0, 1, 0, 0, 1)$;
 - (b) Set $(n(t+1), n(t+2), \dots, n(t+6)) = (3, 2, 1, 3, 2, 3) + (0, 0, 1, 0, 0, 1) = (3, 2, 2, 3, 2, 4)$;
 - (c) Find $n_2 = \max\{3, 2, 2, 3, 2, 4\} = 4$.
 - $i = 1$:
 - (a) Set $c_j(t) = 1$ and find $(w_j(t+1), w_j(t+2), \dots, w_j(t+6)) = (0, 1, 0, 0, 1, 0)$;
 - (b) Set $(n(t+1), n(t+2), \dots, n(t+6)) = (3, 2, 1, 3, 2, 3) + (0, 1, 0, 0, 1, 0) = (3, 3, 1, 3, 3, 3)$;
 - (c) Find $n_1 = \max\{3, 3, 1, 3, 3, 3\} = 3$.
 - $i = 0$:
 - (a) Set $c_j(t) = 0$ and find $(w_j(t+1), w_j(t+2), \dots, w_j(t+6)) = (1, 0, 0, 1, 0, 0)$;
 - (b) Set $(n(t+1), n(t+2), \dots, n(t+6)) = (3, 2, 1, 3, 2, 3) + (1, 0, 0, 1, 0, 0) = (4, 2, 1, 4, 2, 3)$;
 - (c) Find $n_0 = \max\{4, 2, 1, 4, 2, 3\} = 4$.
3. Find $n^* = \min\{4, 3, 4\} = 3$, i.e., $n^* = n_1$, and thus set $c_j(t) = 1$.

Note that according to IEEE 802.11 standard, if mobile station *j* has no data to send, it can send a Null data frame with Power Management bit set to AP. The AP begins buffering frames and sends an ACK frame to the station after receiving the Null data frame. We can just modify this step to incorporate our wakeup scheduling in IEEE 802.11 standard as follows: The AP begins buffering frames, determines $c_j(t)$ and sends an ACK frame with $c_j(t)$ value to station *j* after receiving the Null data frame. Then, the station *j* sets

Suppose there are 2, 2, 1, 2 packets send to station A, B, C, D in each beacon interval

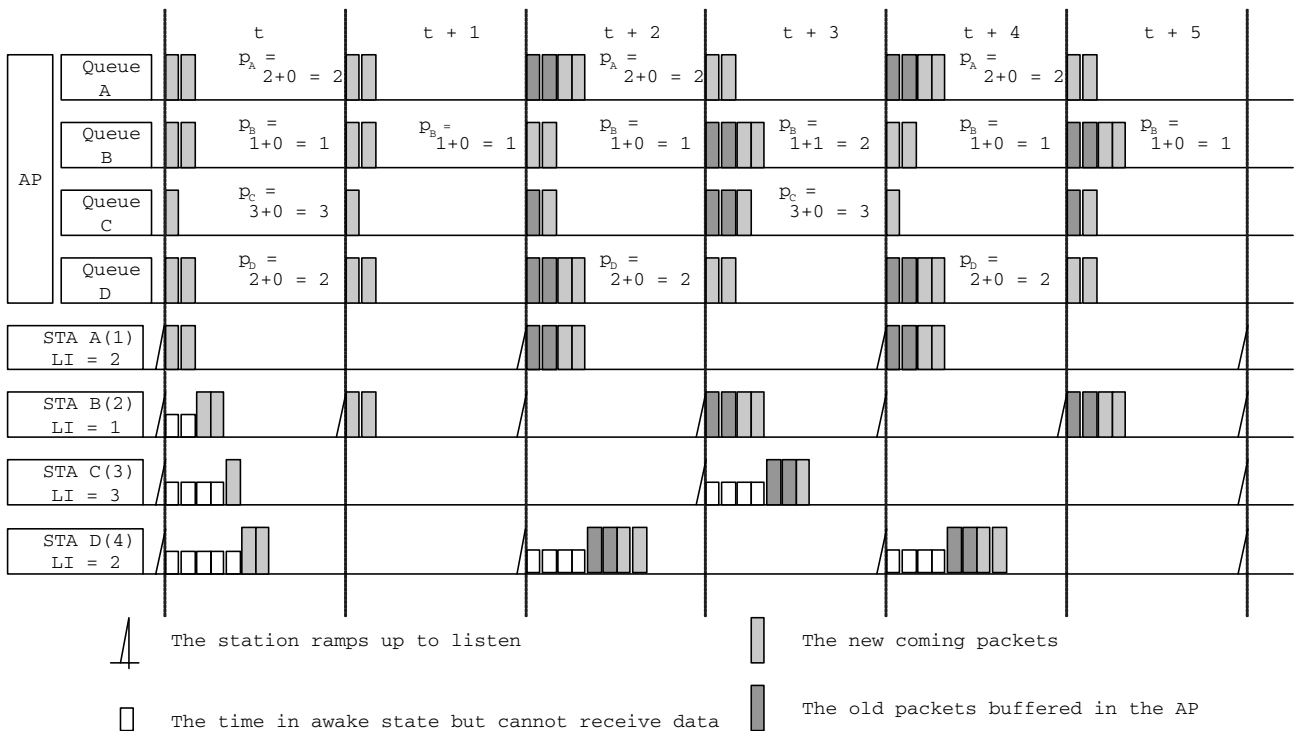


Fig. 3. An example of the smallest AID first method.

its wakeup counter to $c_j(t)$ and enters the sleeping mode.

4. Contention avoidance traffic scheduling

In the previous section, we arrange stations' wakeup times so that the number of wakeup stations is balanced. In this section, we consider how to inform stations that frames are buffered such that the contention is avoided. Three different access scheduling mechanisms are proposed for the contention avoidance problem. In the first mechanism, only one wakeup station is scheduled to access the buffered data in a beacon interval by marking one bit in TIM. The second and third mechanisms schedule multiple wakeup stations to get back their buffered data within a beacon interval. The access sequence within the beacon interval is according to their AIDs and the length of queuing data.

4.1. Multiple wakeups single access

One of simple ways to avoid contention is that we only choose a station to inform it that AP has its buffered frames at each beacon interval. So there is no contention problem of sending PS-Poll frame to get back its buffered data. Let $S_w(t)$ be the set including all stations waking up at beacon interval t . That is,

$$S_w(t) = \{i | i \in S, c_i(t) = 0\}$$

where S is the set including all sleeping stations. Let $S_b(t)$ be the set including all stations that frames are buffered in AP at beacon interval t . Thus, we can choose a station, say station v , from set $S_w(t) \cap S_b(t)$ with a largest listen-interval ℓ_v to inform that the AP has buffered frames for it.

It is possible that there may exist some stations with small listen interval in set $S_w(t) \cap S_b(t)$ and they are never chosen by AP. To avoid such a case, we associate each station v in $S_w(t) \cap S_b(t)$ with an age, denoted as a_v . Initially, the age of each station is set to zero. For each beacon interval, if a station in set $S_w(t) \cap S_b(t)$ is not selected to inform, AP increases its age by one; otherwise, AP sets its age to zero. Thus, AP can choose a station, say station v , from set $S_w(t) \cap S_b(t)$ with a largest value of $\ell_v + a_v$ to inform.

Fig. 2 shows an example of this mechanism. Consider that there are four stations, A, B, C , and D , with listen-interval $(\ell_A, \ell_B, \ell_C, \ell_D) = (2, 2, 3, 1)$. Packet arrival rate of each station is one frame per beacon interval. In beacon interval t , stations A, C , and D wake up in which station C , has maximum $p_C = \ell_C + a_C = 3 + 0$, is indicated in TIM to inform it that the AP has buffered its data. Stations A and D are deferred to their next wakeup beacon intervals. The AP sets ages $a_A = a_A + 1$ and $a_D = a_D + 1$. In the beacon interval $t + 1$, stations B and D wake up. Because $\ell_B + a_B = \ell_C + a_C = 2$, the AP selects station B , arbitrarily, to inform it has buffered frames. Similarly, station A is chosen to inform in beacon interval $t + 2$. At beacon interval $t + 3$, $\ell_B + a_B < \ell_C + a_C < \ell_D + a_D$ and thus station D is chosen to inform.

4.2. Multiple wakeups multiple accesses

Although the multiple wakeups single access mechanism avoids the contention among stations, it may lower the bandwidth utilization and increase the transmission delay. However, the AP knows how many frames it has buffered in queue, transmission rate and the length of beacon interval. Thus, the AP can determine how many frames it can transmit in a beacon interval and schedule the buffered

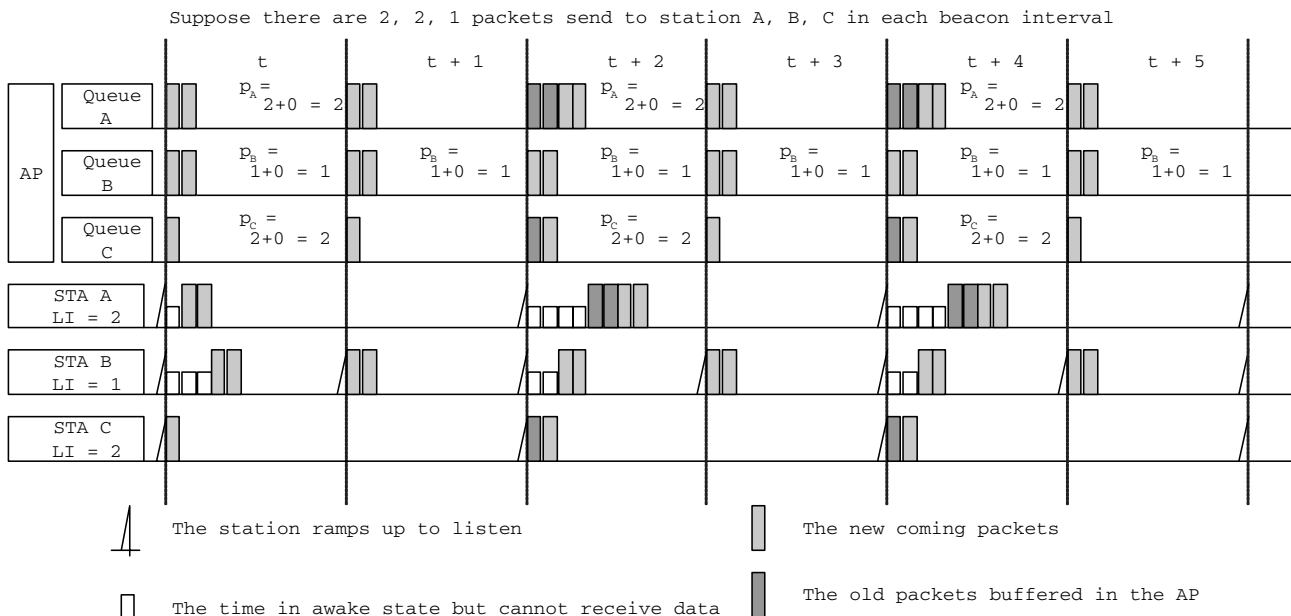


Fig. 4. An example of the smallest queue length first method.

Table 4
Detail simulation configurations

Data rates	11 Mbps
MAC header	28 bytes
IP header	20 bytes
UDP header	20 bytes
Beacon frame	28 bytes
ACK frame	14 bytes
PS-Poll frame	14 bytes
SIFS	0.00001 s
DIFS	0.00005 s
Slot time	0.00002 s
Beacon interval	0.1 s

frame by means of announcing the TIM. In the following, we give two methods to arrange the access sequences of stations.

4.2.1. The smallest AID first

In order to control the traffic load in a beacon interval, AP selects a set of stations with an appropriate size from $S_w(t) \cap S_b(t)$ to inform them to retrieve the data. That is the total amount of buffered frames of selected stations should be less than the capacity of a listen interval. This can avoid a station that awakes within whole beacon interval but can not get back its buffered data. Next, we modify the power management scheme of 802.11 WLAN such that a station retrieves the buffered frame according to the sequence of AID marked in TIM. That is, the station with smallest AID among the selected stations sends PS-Poll frame to retrieve buffered data first.

Fig. 3 shows an example of the AID sequence method. There are four stations, *A*, *B*, *C*, and *D* with listen interval $(\ell_A, \ell_B, \ell_C, \ell_D) = (2, 1, 3, 2)$ with under the service of an AP. Their corresponding AIDs are 1, 2, 3, and 4 for stations *A*,

B, *C*, and *D*, respectively. Suppose packet arrival rates of stations *A*, *B*, *C*, and *D* are 2, 2, 1, and 2 per beacon interval. The maximum size that AP can transfer to stations in a beacon interval is 8 frames. In beacon interval *t*, all of four stations wake up. Because the number of buffered frames is $2 + 2 + 1 + 2 = 7 (7 < 8)$, the AP marks AIDs 1, 2, 3, and 4 in the TIM. The stations check the TIM in beacon frame. They learn that 4 stations will send PS-Poll to retrieve their buffered frames and every station knows which station precedes it in access sequence. For example, station *C* has to wait stations *A* and *B* finishing their access. In beacon interval $t + 2$, $S_w(t + 2) \cap S_b(t + 2) = \{A, B, D\}$ and the number of frames buffered for stations, *A*, *B*, and *D* is 10 ($10 > 8$). Thus, based on the values of p_A and p_D , the AP selects stations *A* and *D* to inform them to retrieve the buffered data.

4.2.2. The smallest queue length first

Instead of the smallest AID first, the AP can arrange the access sequence for the selected stations according to their associated queue lengths. The station with smallest queue length receives a highest precedence and thus it can have a longer sleeping time. In this method, we need to add an information element, describes the access sequence, as a component of the beacon frame. The station checks this information element for the access sequence.

Fig. 4 shows an example of the smallest queue length first method. There are three stations, *A*, *B*, and *C* with listen-interval $(\ell_A, \ell_B, \ell_C) = (2, 1, 2)$ with under the service of an AP. Suppose packet arrival rates of stations *A*, *B*, and *C* are 2, 2, and 1 frames per beacon interval. The maximum size that AP can transfer to stations in a beacon interval is assumed to be 8 frames. In beacon interval *t*, all of the three stations wake up. Because the number of buffered frames are $2 + 2 + 1 = 5 (5 < 8)$, the AP marks AIDs 1, 2, and 3 in the TIM and adds the access sequence *C*, *A*,

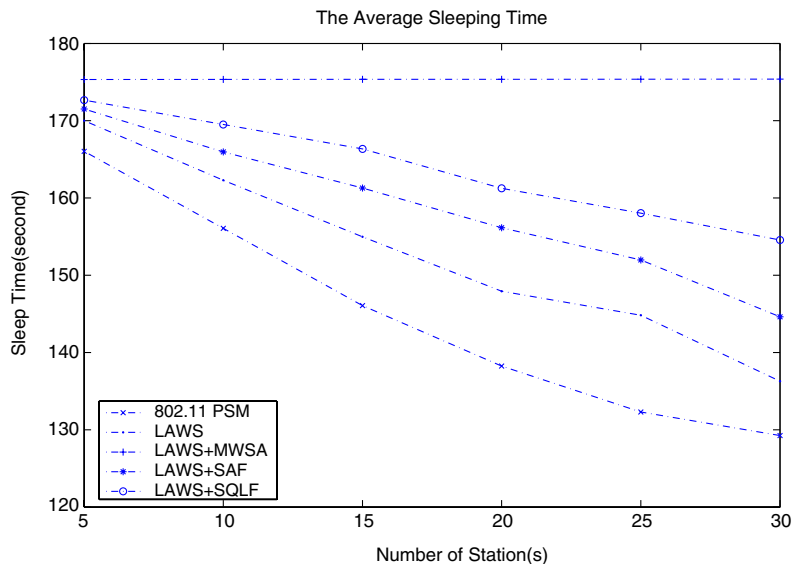


Fig. 5. The average sleeping time.

and B in the beacon frame. In beacon interval $t + 2$, the access sequence is stations C , B , and A . Note that if two stations have same queue length, AP uses their p_v values to break the tie.

5. Simulation and results

5.1. Performance metrics and environment setup

In this section, we show the performance analysis for the proposed schemes:

1. Load-aware wakeup scheduling (LAWS);

2. LAWS with multiple wakeups single access (LAWS + MWSA);
3. LAWS with multiple wakeups multiple access and the smallest AID first (LAWS + SAF);
4. LAWS with multiple wakeups multiple access and the smallest queue length first (LAWS+SQLF).

Note that all four schemes are enhancements of the 802.11 PS mode. The LAWS arranges station's wakeup time. The MWSA, SAF, and SQLF schemes can be used by AP to schedule the access sequence by marking the bits in TIM. We compare their performances against pure 802.11 PS mode by simulation. The performance metrics are given as follows:

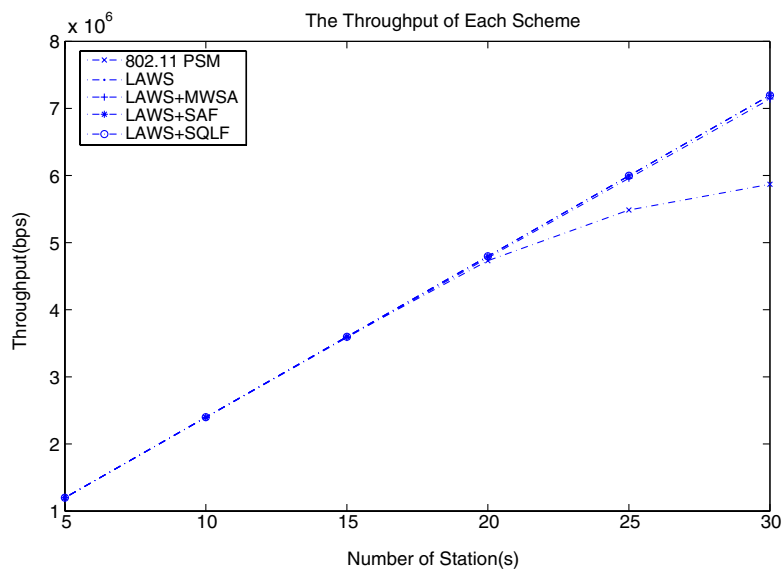


Fig. 6. The average throughput for each scheme.

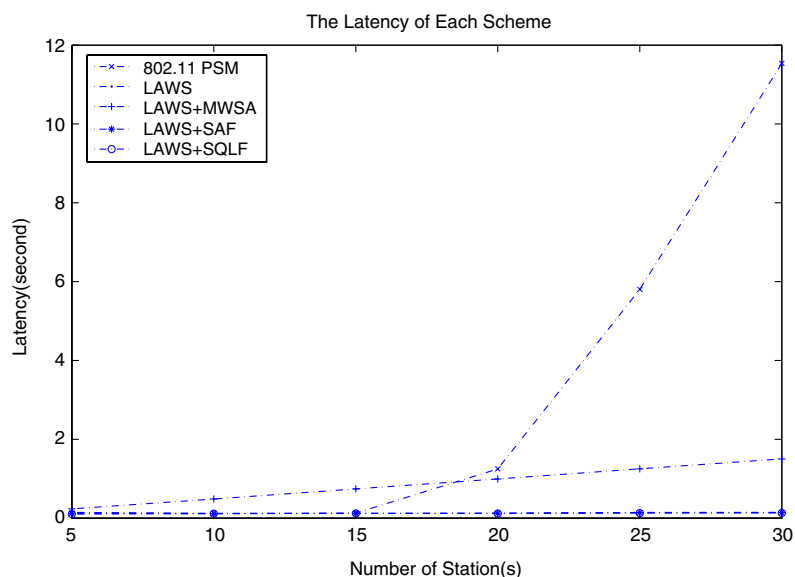


Fig. 7. The latency of a successful transmission for each scheme.

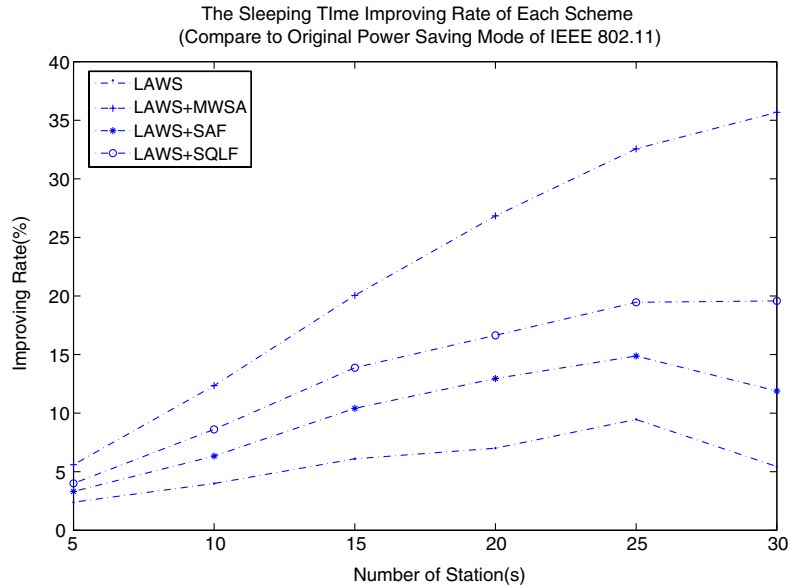


Fig. 8. The improving rate of sleeping time (compared to PS mode of IEEE 802.11).

1. *Average sleeping time of the station:* This measure is the duration that a station stays in the sleeping mode. If a scheme can make stations stay more time in sleeping, then stations will save more power.
2. *Average throughput:* This value shows the total amount of data successfully transmitting per second. If AP can efficiently schedule and distribute the access of its serving stations, it will have higher data throughput.
3. *Average latency of a successful transmission:* The latency is defined as the time duration starting while a packet is issued and buffered at AP and ending when the target station returns the acknowledge. An AP with a good scheduling scheme will make the latency as small as possible. Thus, the resources required for buffering data can be reduced.

Our simulation uses an IEEE 802.11b wireless communication module with 11 Mbps data rate. An AP can serve at most 30 stations. Each station will randomly set 1 to 5 beacon intervals as its listen-interval size and its packet arrival rate is 3 packets per beacon interval. Packet size in our simulation is fixed and set to 1 kbyte. Communication channel assumes to be clear and symmetric. The total simulation time is 3 min. The details of other simulation configurations such as header length, and inter-frame spaces (IFS) are listed in Table 4. Simulation results will compare the IEEE 802.11 PS mode with the proposed LAWS, LAWS + MWSA, LAWS + SAF, and LAWS + SQLF schemes.

5.2. Results and discussion

Fig. 5 shows the relation between average sleeping time and number of stations. Considering contention-based schemes, LAWS can have more sleeping time than IEEE

802.11 PS mode in any size of stations. By using LAWS + MWSA, LAWS + SAF, and LAWS + SQLF schemes to reduce the contention within a beacon interval, stations can have more time on staying in sleeping than LAWS and IEEE 802.11 PS mode. In this figure, it seems that LAWS + MWSA has better sleeping time than both LAWS + SAF and LAWS + SQLF. However, we will find in Fig. 7 that it trades the transmission latency with the sleeping time.

Fig. 6 shows the average throughput for each scheme. From this figure, we can explicitly find that the throughput of IEEE 802.11 PS mode falls down when station number is greater than 20. However, our proposed schemes, LAWS, LAWS + MWSA, LAWS + SAF, and LAWS + SQLF, are not influenced as number of station increases. This is because our schemes can efficiently avoid the data collision among the stations.

In Fig. 7, we show the average latency of a successful transmission for each scheme. For LAWS, LAWS + SAF, and LAWS + SQLF, all of their latency is smaller than 0.3 s and increase slowly as number of stations grows. Because only one station is indicated within a beacon interval, the latency of LAWS + MWSA scheme is longer than the other proposed schemes. The pure IEEE 802.11 PS mode, however, will suffer the worst latency while number of stations increases.

Finally, Fig. 8 shows the improving rate of sleeping time for each proposed scheme (compared to pure IEEE 802.11 PS mode). The improving rate R_i of scheme i is defined as $R_i = \frac{S_i - S_0}{S_0} \times 100\%$, where S_0 and S_i are the average sleeping times for pure 802.11 PS mode and the proposed scheme i , respectively. By efficiently scheduling the wakeup time of sleeping stations, the sleeping duration of LAWS, LAWS + MWSA, LAWS + SAF, and LAWS + SQLF schemes can be improved significantly.

6. Conclusion

In this paper, we propose a load-aware wakeup schedule scheme for infrastructure mode of IEEE 802.11 WLANs. The LAWS scheme balances the number of wakeup stations in each beacon interval to reduce the amount of contention stations. For avoiding the contention, MWSA, SAF, and SQLF scheme are used to arrange the access sequence of the wakeup stations within a beacon interval. Simulation results show that comparing to 802.11 PS-mode, the proposed LAWS, MWSA, SAF, and SQLF schemes can efficiently improve the sleeping duration of each station, average throughput, and transmission delay.

The following two issues should be considered in the implementation of the proposed schemes:

1. An *aging function* should be implemented in the AP to determine when buffered frames are old enough to be discarded.
2. If the mobile station misses the beacon, it should remain awake until it receives the next beacon. The mobile station checks the beacon frame. If the bit corresponding to its AID is set to zero in the TIM, or else it has retrieved all buffered frames, the mobile station can resume the sleeping mode by asking AP for a new wakeup counter $c_i(t)$. In the LAWS + SAF and LAWS + SQLF schemes, the mobile station misses the beacon can not show up to retrieve the buffered data in its turn. The next station in the access sequence can send PS-Poll frames to get back its buffered data if it finds that the medium has been idle for longer than the distributed coordination function inter-frame space (DIFS).

Finally, finding a shorter repeating pattern for LAWS scheme or extending LAWS scheme to wireless ad hoc networks might be interesting for possible future work.

References

- [1] IEEE 802.11 working group. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, (1999).
- [2] IEEE 802.11 working group. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: high-speed physical layer in the 5 GHz Band, (1999).
- [3] IEEE 802.11 working group. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: higher-speed physical layer extension in the 2.4 GHz Band, (1999).
- [4] C. Jones, K. Sivalingam, P. Agrawal, J. Chen, A survey of energy efficient network protocols for wireless networks, *ACM J. Wireless Netw.* 7 (4) (2001) 343–358.
- [5] R. Krashinsky, and H. Balakrishnan, “Minimizing energy for wireless web access with bounded slowdown”, in: Proceedings of the 8th Annual International Conference on Mobile Computing and Networking, Atlanta, Georgia, USA, pp. 119–130 September 23–28, (2002).
- [6] A. Sheth, and R. Han, “Adaptive power control and selective radio activation for low-power infrastructure-mode 802.11 LANs”, in : Proceedings of 23rd International Conference on Distributed Computing Systems Workshops (ICDCSW’03), Providence, Rhode Island, USA, pp. 812–817, May 19–22, (2003).
- [7] J.R. Jiang, Y.C. Tseng, C.S. Hsu, and T.H. Lai, “Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks”, in: Proceedings of International Conference on Parallel Processing, Nice, France, pp. 812–817, April 22–26, (2003).
- [8] J.R. Jiang, Y.C. Tseng, C.S. Hsu, and T.H. Lai, “Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks”, *ACM mobile networking and applications (MONET), special issue on algorithmic solutions for wireless, mobile, ad hoc and sensor networks*, vol. 10, No. 1/2, pp. 169–181, (2005).
- [9] J. Lorchat, T. Noel, “Energy saving in IEEE 802.11 communications using frame aggregation”, *IEEE Global Communication Conference (GLOBECOM’03)*, San Francisco, USA, December, (2003).
- [10] Y. Agarwal, C. Schurgers, and R. Gupta, “Dynamic power management using on demand paging for networked embedded systems”, *Asia South Pacific Design Automation Conference (ASP-DAC’05)*, Shanghai, China, pp. 755–759, January 18–21, (2005).
- [11] C.S. Raghavendra, S. Singh, PAMAS—power aware multi-access protocol with signalling for ad hoc networks, *ACM SIGCOMM Comput. Commun. Rev.* 28 (1998) 5–26.
- [12] J. Snow, W.-C. Feng and W. Feng, “Implementing a low power TDMA protocol over 802.11”, in: Proceedings of IEEE Wireless Communications and Networking Conference, New Orleans, USA, pp. 75–80, March 13–17, (2005).
- [13] J.A. Stine, G. de Veciana, Improving energy efficiency of centrally controlled wireless data networks, *Wireless Netw.* 8 (2002) 681–700.
- [14] M. Liu and M.T. Liu, “A power-saving scheduling for IEEE 802.11 mobile ad hoc network”, in: Proceedings of the 2003 International Conference on Computer Networks and Mobile Computing, Shanghai, China, pp. 1–8, October 20–23, (2003).



Hsiao-Po Lin received the B.S. degree and M.S. degrees in Computer and Information Science from National Chiao Tung University, Taiwan, in 2003 and 2005, respectively. His research interests include wireless networks, mobile computing and internet protocol.



Shih-Chang Huang received the B.S. degree in Computer Science and Engineering from Tatung University, Taiwan, in 1997, and received M.S. degree in Computer Science from National Tsing Hua University, Taiwan, in 1999. Since 2002, he has been working toward the Ph.D. degree in Computer and Information Science at National Chiao Tung University, Taiwan. His research interests include wireless network and wireless sensor network.



Rong-Hong Jan received the B.S. and M.S. degrees in Industrial Engineering, and the Ph.D. degree in Computer Science from National Tsing Hua University, Taiwan, in 1979, 1983, and 1987, respectively. He joined the Department of Computer and Information Science, National Chiao Tung University, in 1987, where he is currently a Professor. During 1991–1992, he was a Visiting Associate Professor in the Department of Computer Science, University of Maryland, College Park, MD. His research interests include wireless networks, mobile computing, distributed systems, network reliability, and operations research.