

M-PSM: Mobility-Aware Power Save Mode for IEEE 802.11 WLANs

Xi Chen[†], Sunggeun Jin^{†‡} and Daji Qiao[†]

[†]Iowa State University, Ames, IA 50011

[‡]ETRI, Daejeon, Korea

{leon6827, sgjin, daji}@iastate.edu

Abstract—With the proliferation of Wi-Fi equipped mobile devices such as smart phones, it becomes even more important to design and implement effective power management schemes for Wi-Fi interfaces so that the battery lifetime can be prolonged. In this paper, we propose an enhancement to the default 802.11 Power Save Mode (PSM), called M-PSM, which exploits additional power-saving opportunities by considering user mobility and detailed traffic condition when making the sleep/wakeup schedules for Wi-Fi interfaces. We have implemented M-PSM in the Madwifi device driver and demonstrated its effectiveness via experiments and trace-based simulations.

I. INTRODUCTION

Recent commercial success of Wi-Fi equipped mobile devices such as smart phones opens a new era that users may rely on Wi-Fi networks to receive broadband services anytime and anywhere in various environments (e.g., static or mobile, indoor or outdoor, etc.). Mobile devices usually are battery-powered and have limited amount of energy, and Wi-Fi radio has been recognized as a prime source of energy consumption in mobile devices. Therefore, it is critical to implement an effective power management scheme for Wi-Fi interfaces.

The 802.11 standard specifies a default power management scheme, called *Power Save Mode (PSM)*, which allows an 802.11 station to sleep for most of the time and only wake up periodically to receive the packets buffered at the Access Point (AP). The PSM was originally designed to reduce the power consumption of a wireless station when the station is lightly-loaded with delay-insensitive traffic. One of the limitations of the default PSM scheme is that it does not consider either user mobility or detailed traffic condition when making the sleep/wakeup decisions. In other words, the station in PSM wakes up at every beacon interval and retrieves the buffered packets (if any) without considering its mobility pattern or the current traffic rate. In fact, more power-saving opportunities may emerge when these factors are considered. For example, by considering the user mobility pattern, more energy may be saved by buffering packets when the user is far away from the AP and transmitting them later at higher rates (hence less time and less energy consumption) when the user moves closer to the AP. Similarly, by considering the traffic condition, more energy may be saved by allowing the station to wake up less frequently (hence less wakeup overheads) and retrieve all the buffered packets during a single beacon interval.

In this paper, we propose an enhanced PSM scheme, called

M-PSM (Mobility-aware Power Save Mode), which exploits these additional power-saving opportunities. Specifically, M-PSM has the following features:

- M-PSM is intended to replace PSM to conserve more energy for a mobile station, particularly when the station is lightly-loaded with delay-insensitive traffic.
- M-PSM estimates the large-scale trend of the channel condition between a mobile station and the AP, based on which it buffers the packets whenever there is a chance for more energy saving.
- M-PSM monitors the traffic rate destined for a mobile station, based on which it adjusts the sleep interval for the station to reduce wakeup overheads.
- M-PSM avoids retrieving the buffered packets during deep fading dips by monitoring small-scale variations of the channel condition.
- We have implemented M-PSM in the Madwifi device driver and demonstrated its effectiveness via experiments.

The rest of the paper is organized as follows: in Section II, we briefly explain the basic operations of the 802.11 PSM and discuss the related work; in Section III, we present a few observations regarding the 802.11 PSM based on experiments and simulations; section IV describes the design and implementation details of the proposed M-PSM scheme. Performance evaluation results are given in Section V; and the paper concludes in Section VI.

II. PRELIMINARY AND RELATED WORK

A. Power Save Mode (PSM) in 802.11 Networks

The 802.11 standard [1] specifies that an 802.11 station may be in one of the two power states: *active* or *doze*. In the active state, the station is fully powered and is ready to communicate with others at any time. In contrast, in the doze state, the station is powered down and consumes extremely low energy. However, it cannot transmit/receive packets or sense the wireless channel. Depending on how a station switches between active and doze states, there are two different operational modes for an 802.11 station: *Active Mode (AM)* or *Power Save Mode (PSM)*. AM Stations stay in the active state all the time, while PSM stations stay in the doze state for most of the time and only wake up at certain time instances to enter the active state to communicate with the AP. We are particularly interested in the PSM in this work.

An 802.11 station enters/leaves the PSM by transmitting data frames with the Power Management (PM) bit set to 1/0. The AP keeps track of the operational modes of all the stations that are associated with it. When a station is in PSM, the AP buffers the packets that are destined for the station and announces the buffer status every Target Beacon Transmission Time (TBTT) via the Traffic Indication Map (TIM) carried in the Beacon frame. If the TIM indicates the presence of buffered packets at the AP for a station, the station stays in the active state and sends PS-Poll messages to retrieve the buffered packets, one at a time, till all the packets are received; otherwise, it goes back to sleep in the doze state.

B. Related Work on PSM

Over the years, much work has been proposed to improve the performance of the 802.11 PSM. Unfortunately, little has been done to exploit how the user mobility may help improve PSM and how a PSM station shall behave when it is mobile.

In [2], the authors proposed to use a special hardware (a small radio device) to help reduce the energy consumption of an 802.11 station by allowing the station to stay in the deep sleep state and only waking it up when new data services for the station have been initiated by the AP. The authors of [3], [4] studied the tradeoff between minimizing energy consumption and reducing response delay with the 802.11 PSM. But, user mobility was not considered in their studies.

In [5]–[7], the authors proposed new protocols to minimize the energy consumption of a PSM station in the deep sleep state even when it moves across AP cell boundaries. Additionally, they found that, in order to retrieve the broadcast/multicast packets of its own interests, a PSM station has to receive the entire broadcast/multicast traffic from the AP, called *background traffic*, involuntarily since it cannot distinguish between packets *a priori* before the reception. Clearly, the presence of the background traffic may deteriorate the station’s power saving efficiency. The authors of [8], [9] analyzed this issue in detail and suggested a new operation to mitigate the impact while maintaining the backward compatibility with the 802.11 PSM. In [10], the authors proposed a scheduling policy to reorder traffic transmission priorities according to the operational states of destination stations. The key idea is that the AP transmits the traffic destined for PSM stations prior to the traffic for AM stations.

The proliferation of Wi-Fi equipped smart phones promotes the researches that target improving the power saving efficiency of an 802.11 PSM station when it is actively serving Voice over IP (VoIP) sessions [11]–[15]. In [16]–[18], the authors proposed to utilize proxies to aggregate traffic, thus reducing transmission overhead and energy consumption. Basically, the proxies reside in the AP, buffer the traffic destined for each PSM station until the buffered traffic could mitigate the transmission overhead, and then forward the buffered traffic with proper aggregation. The authors of [19] studied the relationship between user mobility and power consumption in cellular networks. They proposed a history-based energy conservation scheme for cellular devices, which

requires repetitive training before usage and hence may not be suitable for dynamic environments where routes are not known *a priori*.

III. OBSERVATIONS

In this section, we present a few interesting observations regarding the 802.11 PSM from experiments and simulations, which motivate us to design our proposed scheme. Experiments are conducted with Dell Latitude laptops equipped with Linksys WPC55AG cards. Each laptop is loaded with the Madwifi device driver v0.9.4 [20] to collect experimental data. Simulations are conducted in Matlab using the traces collected from the experiments.

A. Effects of Buffering

Rate adaptation is a well-known technique to improve the throughput between an 802.11 station and the AP. The idea is to exploit the multiple transmission rates available for an 802.11 station and adjust its transmission rate dynamically to the time-varying and location-dependent channel condition between the station and the AP. With an effective rate adaptation scheme, a data packet may be transmitted at the highest rate that can be accommodated by the channel condition; hence less transmission time is required and less energy is consumed. This implies that the power saving efficiency of the default PSM scheme could be improved with a smart buffering strategy. The idea is that the AP buffers packets when the channel condition is bad and the station issues PS-Polls to retrieve the buffered packets only when the channel condition has improved enough to accommodate higher transmission rates. Combining buffering with PSM could be particularly beneficial for mobile stations, which may observe large-scale fluctuations of the channel condition.

Fig. 1 plots the simulation results of PSM with a simple buffering strategy, with which the station issues PS-Polls only when the SNR is higher than 33 dB. Here, 33 dB is the minimum SNR value we found in experiments that data packets can be transmitted at the highest rate of 54 Mbps with negligible packet loss.

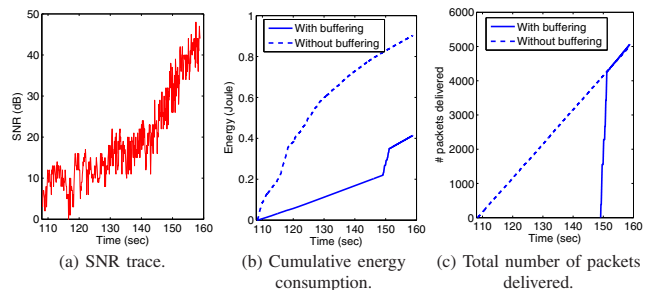


Fig. 1. Performance of PSM with a simple buffering strategy.

Fig. 1(a) shows the SNR trace used in the simulation, which is collected with a user carrying a laptop and walking towards the AP. Fig. 1(c) shows that, with the simple buffering strategy, the AP starts to deliver the buffered packets around the 149-second mark when the channel condition becomes very good.

Comparing with the default PSM scheme, the extra buffering between the 109-second and 149-second marks yields 56% less energy consumption at the end of the simulation (i.e., 0.42 Joule with buffering vs 0.95 Joule without buffering at the 159-second mark). However, it should be noticed that such a simple buffering strategy might not work in other situations. For example, if the current SNR is lower than 33 dB but the channel condition is deteriorating (e.g., when the user is walking away from the AP), the buffered packets may never get a chance to be transmitted. So it is important to design a smart buffering strategy to work with PSM so that more energy may be conserved while all the packets are still delivered. The buffering strategy used in our proposed power-saving scheme will be discussed in Section IV.

B. Wakeup Overhead

To guarantee reception of the Beacon frames from the AP, an 802.11 station usually needs to wake up a bit earlier than the expected beacon arrival time. We call this lead time the *wakeup overhead*. The main reasons for the need of a wakeup overhead are that (i) the radio switching takes certain amount of time; and (ii) time asynchrony between the AP and the station due to clock drift. We have done some experiments to study the effects of the wakeup overhead. Fig. 2 plots the relationship between the wakeup overhead and the Beacon frame loss ratio for Linksys WPC55AG cards. Similar trend can be found for other Wi-Fi cards.

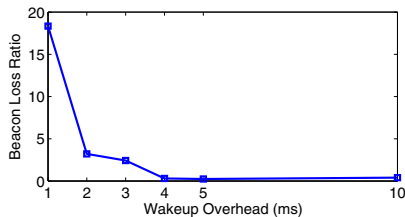


Fig. 2. Effects of wakeup overhead on beacon reception for Linksys WPC55AG cards.

In general, the station experiences a higher beacon loss ratio with a smaller wakeup overhead. As shown in the figure, the beacon loss ratio could be as high as 18% when the wakeup overhead is only 1 ms. From the experiments, we can see that a minimum of 4 ms wakeup overhead is needed to guarantee a high beacon reception ratio. One way to reduce the overall wakeup overhead for an 802.11 station is that the station may skip some of the beacon announcements when the traffic is light, and wake up later to retrieve all the buffered packets from the AP with a single wakeup overhead.

C. Effects of Beacon Monitoring Interval (BMI)

As discussed above, an 802.11 station may skip some of the beacon announcements to conserve energy. We call the interval between two adjacent time instances when the station wakes up to receive beacon announcements as the *Beacon Monitoring Interval (BMI)*. For simplicity, we represent BMI in units of beacon intervals (i.e., 100 ms by default). So in a conventional

802.11 network, BMI = 1, meaning that the station wakes up every beacon interval to listen for all beacon announcements from the AP.

We have performed trace-based simulations with different BMI values and results are plotted in Fig. 3. As shown in the figure, with a larger BMI, more packets are queued at the AP. As a result, the queue length increases and so does the packet delivery delay. On the other hand, more energy is conserved because more wakeup overheads are saved, particularly when the traffic rate is low (e.g., $\lambda = 0.01$ packets/ms). So there is a tradeoff in setting the proper BMI. We will discuss how BMI is adaptively adjusted in our proposed scheme in Section IV.

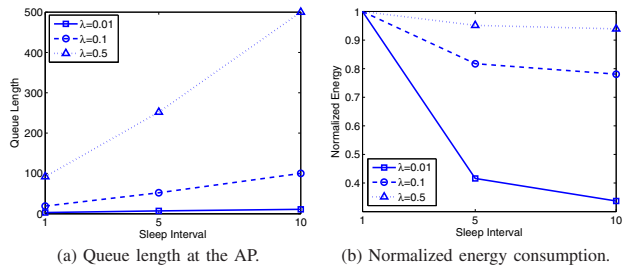


Fig. 3. Effects of Beacon Monitoring Interval (BMI). λ is the traffic rate (in units of packets/ms).

D. Design Guidelines

Based on the above observations, we conclude that an enhanced PSM scheme shall have the following features.

- It shall be able to predict the large-scale trend of the channel condition, and determine possible long-term buffering opportunities to fully utilize the high transmission rates in the future.
- It shall be able to detect possible short-term buffering opportunities to save the wakeup overhead, while guaranteeing that all the buffered packets can be delivered efficiently upon future wakeup.
- It shall be aware of the queue length as well as the traffic rate, and be able to adjust the BMI adaptively by considering these context information jointly.

IV. M-PSM DESIGN AND IMPLEMENTATION

Following the design guidelines given in Section III-D, we propose an enhancement to the default PSM scheme, called *M-PSM (Mobility-aware Power Save Mode)*. The highlights of the proposed M-PSM scheme are a smart buffering scheme and an adaptive BMI adjustment scheme. In the following, we first introduce the overall structure of M-PSM, and then describe the AP and station behaviors in detail.

A. Overview

The overall structure of the proposed M-PSM scheme is shown in Fig. 4. In M-PSM, the AP buffers the incoming packets for all the stations that are currently associated with it and are in the power save mode. In the periodic Beacon frame sent by the AP, the AP announces the number of buffered packets for each station, in addition to the Traffic Indication

Map (TIM) that only indicates the presence of the buffered packets. Upon reception of a PS-Poll frame from a station, the AP delivers a buffered packet to it. Details of the AP behaviors will be discussed in Section IV-B.

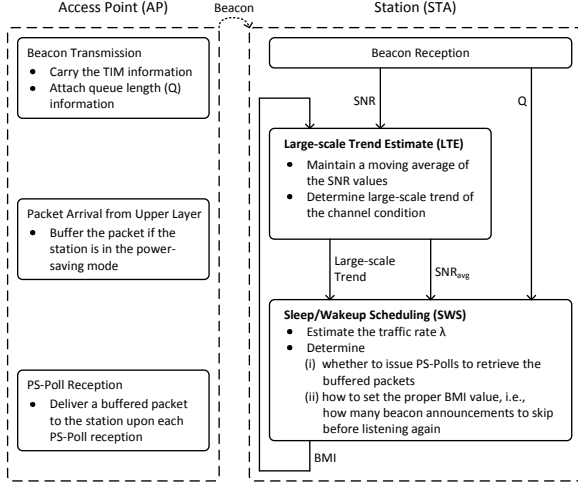


Fig. 4. Overall structure of M-PSM.

In M-PSM, the station in the power save mode executes the following modules upon a beacon reception:

- **Large-scale Trend Estimation (LTE).** This module takes the previously-calculated Beacon Monitoring Interval (BMI) and the SNR of the Beacon frame as inputs. The outputs are the moving average of the SNR values as well as the large-scale trend of the channel condition between the station and the AP, which could be “DOWN”, “STABLE” or “UP”. If the station is mobile, the large-scale trend could indicate the mobility pattern of the station. For example, a “DOWN” large-scale trend of the channel condition could imply that the station is moving away from the AP. Details of the LTE module will be discussed in Section IV-C1.
- **Sleep/Wakeup Scheduling (SWS).** This module accepts the outputs of the LTE module as well as the queue length information carried in the Beacon frame (i.e., the number packets buffered at the AP for the station and denoted by Q). Essentially, this module monitors the traffic rate λ destined for the station, and determines (i) whether to issue PS-Polls to retrieve the buffered packets; and (ii) how long to sleep before waking up for the next beacon announcement, i.e., the next BMI value. Details of the SWS module will be discussed in Section IV-C2.

B. AP Behaviors

Similar to the default PSM, the AP in M-PSM buffers the packets for stations that are in the power save mode and deliver the buffered packets upon receiving PS-Poll messages from stations. The AP also indicates whether there is any packet buffered for the station via the TIM field in the periodic beacon announcement.

In addition, the AP in M-PSM attaches the queue length information in the Beacon frame. The queue length information is used by the station to estimate the traffic rate as well as to plan future sleep/wakeup schedules. In our implementation, we add one extra information element (using one of the reserved Element IDs) to the Beacon frame for this purpose. The format of the information element is shown in Fig. 5. For each station in the power save mode, its buffer queue length and its association ID (AID) are added to the information element. After receiving a Beacon frame, the station looks for its own AID in the information element to make sure that the correct queue length information is obtained. The 802.11 standard specifies that the maximum AID value is 2007. Hence we use 11 bits to contain the AID, and 13 bits to contain the queue length information (i.e., 8191 in maximum).

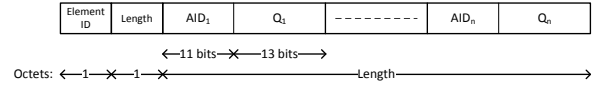


Fig. 5. Format of the information element added to the Beacon frame in M-PSM. n is the number of stations that are in the power save mode.

C. Station Behaviors

In this section, we introduce the two M-PSM modules residing at the station side: the *Large-scale Trend Estimation (LTE)* module and the *Sleep/Wakeup Scheduling (SWS)* module.

1) **Large-scale Trend Estimation (LTE):** The LTE module maintains a moving average of the SNR values of the received Beacon frames, and determines the large-scale trend (LT) of the current channel condition between the station and AP.

1.1) **Calculation of the SNR Moving Average.** Let S_{curr}^t denote the SNR value of the Beacon frame received at time index t . The SNR moving average is calculated as:

$$S_{\text{avg}}^t = \alpha(LT, S_{\text{curr}}^t) \cdot S_{\text{avg}}^{t-1} + [1 - \alpha(LT, S_{\text{curr}}^t)] \cdot S_{\text{curr}}^t, \quad (1)$$

where α is the smoothing factor:

$$\alpha(LT, S_{\text{curr}}^t) = \begin{cases} 0.8, & \text{if } LT == \text{DOWN or STABLE,} \\ \max\left\{0.8, 1 - 0.2 \cdot \frac{S_{\text{curr}}^t}{S_{\text{dB}}^t}\right\}, & \text{if } LT == \text{UP.} \end{cases} \quad (2)$$

The reason for choosing such a smoothing factor is that, when the channel condition is stable or deteriorating, we would like to have S_{avg} track the channel variation more closely so that the station may retrieve the data packets whenever there is a chance; hence a relatively large weight (i.e., 0.2) is given to the new SNR sample. On the other hand, when the channel condition is improving, particularly when the current condition is poor, we would like to have a more conservative S_{avg} estimation which will later be used to predict the large-scale trend of the channel condition; an incorrect large-scale trend prediction could trigger improper buffering decisions and consequently unnecessary transmission delays and failures.

In M-PSM, the station may skip some of the beacon announcements, i.e., $BMI > 1$. In such cases, the moving averages of the SNR values at the skipped beacon times are

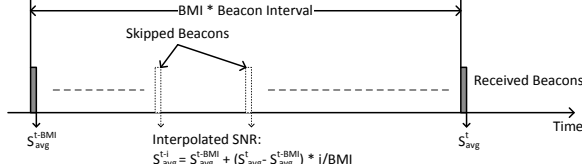


Fig. 6. SNR interpolation in M-PSM.

estimated via interpolation. For example, as shown in Fig. 6, after skipping $(BMI - 1)$ beacon announcements, the station receives a Beacon frame at time index t . Then, we calculate S_{avg}^t as

$$S_{avg}^t = \alpha (LT, S_{curr}^t)^{BMI} \cdot S_{avg}^{t-BMI} + [1 - \alpha (LT, S_{curr}^t)^{BMI}] \cdot S_{curr}^t \quad (3)$$

and estimate S_{avg}^{t-i} ($1 \leq i \leq BMI - 1$) as:

$$S_{avg}^{t-i} = S_{avg}^{t-BMI} + (S_{avg}^t - S_{avg}^{t-BMI}) \cdot \frac{i}{BMI}, \quad (4)$$

This equation will be used next in the pairwise comparison for estimating the LT.

1.2) Pairwise Comparison. The key function of the LTE module is to estimate the large-scale trend of the channel condition between the station and the AP. This is done via simple pairwise comparisons of the previously-calculated or interpolated S_{avg} values. Specifically, suppose a Beacon frame is received at time index t , then the following ϕ comparisons will be performed:

$$(S_{avg}^t - S_{avg}^{t-\phi}), (S_{avg}^{t-1} - S_{avg}^{t-\phi-1}), \dots, (S_{avg}^{t-\phi+1} - S_{avg}^{t-2\phi+1}).$$

If more than ϕ_u out of the ϕ comparisons are larger than 0, we say that LT at time index t is “UP”. On the other hand, if less than ϕ_d out of the ϕ comparisons are smaller than 0, we say that LT at time index t is “DOWN”. Otherwise, LT = STABLE. In M-PSM, we heuristically set $\phi = 10$, $\phi_u = 0.7\phi$, and $\phi_d = 0.3\phi$ based on experimental trials. Table I gives an example of pairwise comparison at time index $t = 20$.

TABLE I
AN EXAMPLE OF PAIRWISE COMPARISON IN THE LTE MODULE

time index t	1	2	3	4	5	6	7	8	9	10
S_{avg}^t (dB)	8.4	7.1	6.1	6.3	6.6	6.9	6.9	6.9	6.1	6.1
Interpolated?	N	N	N	N	N	N	N	N	N	N
time index t	11	12	13	14	15	16	17	18	19	20
S_{avg}^t (dB)	5.5	6.0	7.2	8.7	10.0	11.3	11.3	11.3	11.3	11.3
Interpolated?	N	N	N	N	Y	N	Y	Y	Y	N
Comparison	-	-	+	+	+	+	+	+	+	+
LT	8 of 10 comparisons are positive while $\phi_u = 0.7\phi = 7$; hence LT at $t = 20$ is “UP”.									

At last, we show in Fig. 7 an example trace of outputs by the LTE module collected from one of the experiments. In the figure, the y-coordinates of the markers represent the S_{avg} values at the corresponding time instances. Different markers represent different large-scale trends. For example, at time index $t = 9$ and 16, we have $S_{avg}^9 = 6.1$ dB and $S_{avg}^{16} = 11.3$ dB, LT = DOWN and UP, and BMI = 1 and 4, respectively. Note

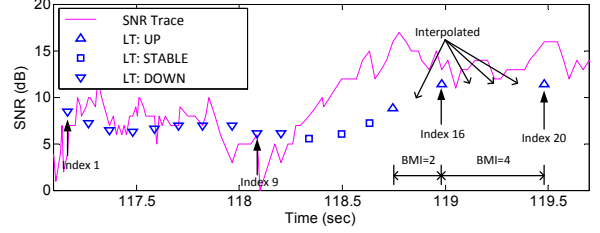


Fig. 7. An example trace of outputs by the LTE module. The pairwise comparison for updating the LT at time index $t = 20$ is explained in Table I.

that, since the LTE module estimates the large-scale trend of the channel condition, its output may not match well with small-scale channel variations, as can be seen in Fig. 7.

2) Sleep/Wakeup Scheduling (SWS): In M-PSM, the SWS module collects the following information from the Beacon frame as well as the LTE module: (i) Q – the queue length information carried in the Beacon frame; (ii) S_{avg} – the moving average of the SNR values; and (iii) LT – the large-scale trend estimated by the LTE module. Based on the collected information, the SWS module performs the following functionalities:

- to estimate of the traffic rate destined for the station;
- to determine whether to issue PS-Polls immediately to retrieve the buffered packets;
- to decide the next BMI value, i.e., the number of beacon announcements to skip, if any.

The goal of the SWS module is to make sure that all the packets destined for the station are retrieved in more energy-efficient manners, i.e., at higher transmission rates with lower wakeup overheads.

The SWS module estimates the traffic rate (λ , in units of packets/ms) at time index t as follows:

$$\lambda^t = \beta \cdot \lambda^{t-1} + (1 - \beta) \cdot \frac{\Delta Q}{\Delta T}, \quad (5)$$

where ΔQ and ΔT are increments of queue length and time since the last beacon reception, respectively. β is the smoothing factor and set to 0.9 in M-PSM.

To determine whether to issue PS-Polls and how to set the next BMI value, the station exploits the following three types of opportunities. Upon each wakeup time, depending on the combination of the available opportunities, the station makes different decisions, as listed in Table II. Next, we elaborate how each opportunity may be detected by the station and the actions taken by the station upon detection.

- *Sleep Opportunity:* Sleep opportunity is an opportunity that allows packet buffering in a relatively long term, in order to exploit the possibly better channel condition in the future. It occurs when the following conditions are satisfied: (i) the AP allows more packets to be buffered for the station; (ii) the large-scale trend is “UP” so buffering packets could be beneficial since the station may be able to transmit the packets later at higher rates when the channel condition improves; and (iii) it is possible that the channel condition may be improved further in the near

TABLE II
DECISION MAKING IN THE SWS MODULE

Case	Conditions	PS-Poll?	How to set BMI?	
1	$Q \geq 0.9 \cdot Q_{\text{limit}}$	Y	BMI = 1	
2	$Q = 0$	N	BMI = $\min\{\text{BMI}_{\text{max}}, 2 \cdot \text{BMI}\}$	
3	$0 < Q < 0.9 \cdot Q_{\text{limit}}$	N	BMI = $\min\{\text{BMI}_{\text{max}}, \text{BMI}_{\text{nap}}\}$ based on Eq. (7)	
4	$\&\& S_{\text{avg}} \geq 33$ dB	Y	BMI = 1	
5	$0 < Q < 0.9 \cdot Q_{\text{limit}}$ $\&\& S_{\text{avg}} < 33$ dB	LT == UP	sleep opportunity	
6		no sleep opportunity	Y	BMI = 1
7	DOWN or STABLE	nap opportunity	N	BMI = $\min\{\text{BMI}_{\text{max}}, \text{BMI}_{\text{nap}}\}$ based on Eq. (7)
8		no nap opportunity	Y	BMI = 1
9		small-scale opportunity	Y	BMI = 1
		no small-scale opportunity	N	BMI remains unchanged

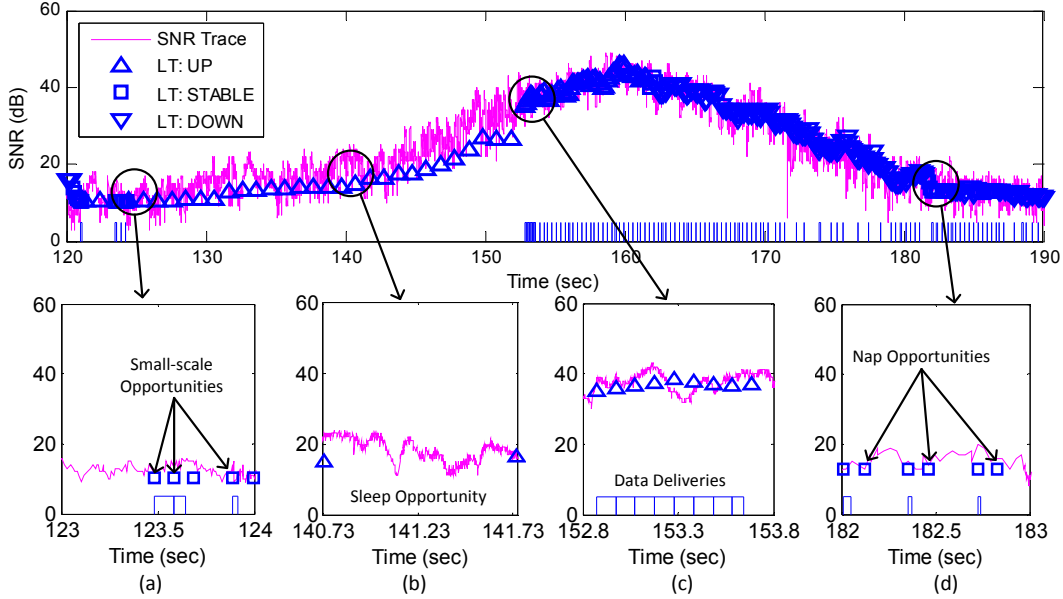


Fig. 8. An example trace of outputs by the SWS module. (a) Example of small-scale opportunities. (b) Example of sleep opportunities. (c) Example of retrieving packets under excellent channel conditions. (d) Example of nap opportunities.

future. In our experiments, 33 dB is the minimum SNR value we found that accommodates transmissions of data packets at the highest rate of 54 Mbps with negligible packet loss. Hence, there is no need for buffering if the current channel condition $S_{\text{avg}} \geq 33$ dB.

To determine the sleep duration, the station needs to make sure that the available queue space at the AP allows more packets to be buffered, i.e., $\text{BMI}_{\text{sleep}} \geq 1$, where

$$\text{BMI}_{\text{sleep}} = \arg \max_k (Q + \lambda \cdot k \cdot \text{Beacon_Interval} \leq 0.9 \cdot Q_{\text{limit}}). \quad (6)$$

Q_{limit} is the maximum number of packets that can be buffered at the AP for the station, which is limited by the memory allocated by the AP to serve the station. Eq. (6) indicates that, after sleeping for $\text{BMI}_{\text{sleep}}$ beacon intervals, the queue length at the AP will not exceed Q_{limit} . As shown in Table II, Case 5, if a sleep opportunity is detected, the station sets a proper BMI value and goes to sleep; otherwise, it issues PS-Polls to retrieve the packets. In practice, different applications may have different delay requirements. To handle this issue, M-PSM provides an option for upper-layer applications to specify their own Q_{limit} values. For example, a Skype application may set a

smaller Q_{limit} value to avoid buffering for too long time.

- *Nap Opportunity*: Nap opportunity is similar to sleep opportunity in that the station tries to exploit the chances to buffer more packets. However, the difference between them is that nap opportunities are shorter-term opportunities and the goal is to save the wakeup overhead. It may present itself only when the large-scale trend is “DOWN” or “STABLE”, or when the channel condition is already good enough (i.e., $S_{\text{avg}} \geq 33$ dB) to accommodate the highest transmission rate. In these situations, although it is unlikely that the station may be able to retrieve the buffered packets in the future at higher rates, more buffering could still allow the station to reduce the wakeup overheads by retrieving many packets together in one single wakeup. Comparing with sleep opportunity, nap opportunity usually is a much shorter-term phenomenon. When exploiting the nap opportunities, the station needs to make sure that all buffered packets could be transmitted during a single beacon interval, i.e., $\text{BMI}_{\text{nap}} \geq 1$, where

$$\text{BMI}_{\text{nap}} = \arg \max_k (Q + \lambda \cdot k \cdot \text{Beacon_Interval} \leq 1 \text{ Mbps} \cdot \text{Beacon_Interval}). \quad (7)$$

Eq. (7) guarantees that after sleeping for BMI_{nap} beacon

intervals, all the buffered packets can be delivered within one single beacon interval, even at the lowest transmission rate of 1 Mbps. For this reason, nap opportunities usually occur when the traffic rate is low. Cases 3 and 7 in Table II show the actions taken by the SWS module when a nap opportunity is detected.

- *Small-scale Opportunity*: Different from the previous two, small-scale opportunity refers to the chances that the station should issue PS-Polls to retrieve the buffered packets. The goal is to, when there are no sleep or nap opportunities, avoid retrieving the packets during deep fading dips, which usually are indicated by a smaller S_{curr} value than S_{avg} . In other words, the station only tries to retrieve packets when $S_{\text{curr}} \geq S_{\text{avg}}$, which is labeled as a small-scale opportunity. This can be seen from Cases 8 and 9 in Table II.

There are two other cases in Table II that are not covered in the previous discussion: Case 1 when $Q \geq 0.9 \cdot Q_{\text{limit}}$ and Case 2 when $Q = 0$, which are straightforward to understand. Basically, if the queue is almost full, the station needs to retrieve immediately regardless of other conditions, while when there is no traffic for the station, the station could sleep longer by skipping more beacon announcements. In M-PSM, BMI_{max} is set to 10, meaning that the station needs to wake up at least once every 1 second to hear the beacon announcements from the AP and keep itself updated of the network/traffic conditions.

We show in Fig. 8 an example trace of outputs by the SWS module, as well as zoomed views at four different time periods. Y-coordinates of the markers represent the S_{avg} values at corresponding time instances. Short bars at bottom of the figures represent that the station is actively retrieving the packets. Sleep or nap opportunities can be observed clearly from (a), (b) and (d), where some of the beacon announcements are skipped to allow more packets to be buffered at the AP. In particular, as shown in (b), the station takes full advantage of the sleep opportunities by buffering all the packets which are transmitted later at higher rates when the channel condition becomes excellent around the 152.9-second mark. Comparing (b) and (d), we can see that nap opportunities are in a smaller scale than sleep opportunities. (a) shows how the SWS module exploits the small-scale opportunities. Upon receptions of the three Beacon frames pointed out in the figure, the station finds that no nap opportunities exist but the instant SNR is higher than the moving average. This means that small-scale opportunities appear and thus the station issues PS-Polls to retrieve packets after receiving these Beacon frames. (c) simply shows that when the channel condition is very good and the traffic rate is high, the station wakes up for every beacon announcement and retrieves as many packets as possible.

Note that Fig. 8 is just an example of the SWS module behavior, and it does not imply that data can only be delivered when the SNR is larger than 33 dB. In fact, as shown in Table II, the decision of when to issue PS-Poll to retrieve data is based on a combination of the current SNR value, the

existence of sleep/nap/small-scale opportunities, and the Q_{limit} value specified by the application.

V. PERFORMANCE EVALUATION

In this section, we evaluate the effectiveness of M-PSM using both experiments and trace-based simulations. Note that in M-PSM, the AP maintains a separate queue for each mobile station; the mobile stations make their own wakeup/sleep scheduling independently, and their actions do not affect the performances of each other. Hence, the performance of M-PSM is not affected by the number of stations in the network. For this reason, the performance evaluation in this work only involves an AP and a mobile station.

A. Experimental Study

1) *Experimental Setup*: We have implemented the proposed M-PSM scheme in Madwifi [20]. Hardware and software configurations used in our experiments are listed in Table III. All the experiments are conducted with Dell Latitude E5400 laptops equipped with Linksys WPC55AG WLAN adaptors, which embed Atheros 5212 chipsets. We use off-the-shelf hardware instead of sophisticated equipments to conduct experiments as this makes our experimental results comparable to what users of commodity 802.11 devices may expect in realistic scenarios. The rate adaptation scheme used in our experiments is the RAM scheme proposed in [21], which is a fast-responsive rate adaptation scheme that represents the state of the art. We use a modified Iperf [22] as the UDP packet generator. CBR (Constant Bit Rate) traffic is generated with the packet size of 1000 octets. We set Q_{limit} to 800 in M-PSM.

TABLE III
CONFIGURATION PARAMETERS

Parameters	Values
Computer	Dell Latitude E5400 Laptop
Operating system	Linux Kernel 2.6.24-16
WLAN adaptor	Linksys WPC55AG
Device driver	Madwifi v0.9.4
PHY	IEEE 802.11g
Transmit Power	17 dBm
CBR packet size	1000 octets

We conduct experiments in both static and mobile scenarios. The experiment venue, the location of the static station as well as the trajectory of the mobile station are shown in Fig. 9. In order to minimize potential unexpected performance variation caused by people's movement and interference from other 802.11, ZigBee and Bluetooth devices, all experiments are conducted at nighttime or weekends. The result for each scenario is averaged over five experimental runs. We compare the performance of the proposed M-PSM scheme against the default 802.11 PSM scheme.

2) *Mobile Scenario*: In the mobile scenario, the station first moves toward the AP and then comes back to the starting point, as shown in Fig. 9, at a walking speed of about 1 m/s. In each experimental run, we measure the amount of time that the station sleeps in the doze state or stays awake in the active state. Fig. 10(a) compares the sleep/active time

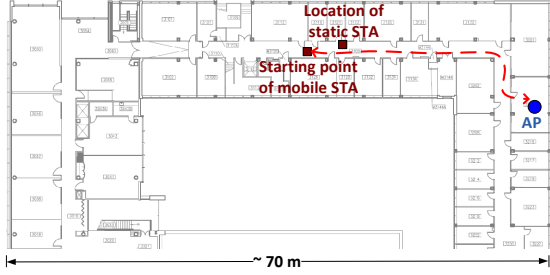


Fig. 9. Experiment venue.

distributions of testing schemes under different traffic rates. Since the Linux kernel does not provide a direct measure of the energy consumption of Wi-Fi cards, we translate the measured time distribution to the energy-efficiency performance using typical Wi-Fi cards' power consumption rates that we find in a recent article [23]: 140 mW in the active state receiving data and 175 μ W in the doze state sleeping. Fig. 10(b) compares the energy efficiency results. Since we are more interested in understanding how M-PSM saves energy via adjusting dynamically to the mobility pattern and traffic condition, rather than the exact amount of energy savings, this translation has limited impact on the observations to be presented next.

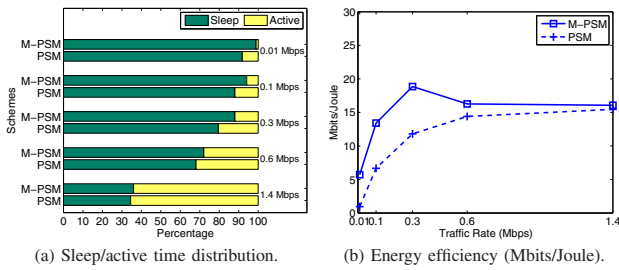


Fig. 10. Comparison of sleep/active time distributions and energy-efficiency performances under different traffic rates. Each point is averaged over five experimental runs.

Firstly, when the traffic rate is low, M-PSM allows the station to sleep for a much longer time than PSM. As shown in Fig. 10(a), when the traffic rate is 0.01 Mbps, the PSM station spends about 8.1% of the time in the active state, while the M-PSM station spends only about 1.1% of the time being active, thanks to M-PSM's capability of adjusting BMI dynamically to the traffic condition. When the traffic rate is low, M-PSM skips most beacon announcements to conserve energy. As a result, the energy efficiency of M-PSM is significantly higher than that of PSM (e.g., about 8 times higher when the traffic rate is 0.01 Mbps), as shown in Fig. 10(b).

Secondly, when the traffic rate is high (i.e., higher than 0.6 Mbps), the energy efficiency of M-PSM is comparable to that of PSM, and becomes stabilized to be around 16 Mbits/Joule. This is because the wireless link between the station and the AP becomes saturated when the traffic rate is high. Thus, there are few sleep or nap opportunities for the station, which spends most of the time retrieving the buffered

packets. In fact, when the traffic rate is high, the energy-efficiency performances of both M-PSM and PSM are similar to that of the Active Mode, which implies that neither M-PSM nor PSM should be used. This indeed conforms to the original design goal of PSM (and hence M-PSM) that is to reduce the power consumption of a wireless station when the station is lightly-loaded with delay-insensitive traffic.

Thirdly, as the traffic rate varies from low to high, the energy efficiency of M-PSM first increases, peaks when the traffic rate is about 0.3 Mbps, and then becomes stabilized. This interesting "peaking" phenomenon is unique and can only be observed in the mobile scenario for the following reason. During the station's movement towards the AP, sleep opportunities appear. Therefore, as the traffic rate increases, more packets may be buffered during each sleep opportunity and get transmitted later at much higher rates. On the other hand, as the traffic rate increases further, the packets get queued up quickly at the AP. As a result, the number of sleep opportunities decreases. The tradeoff between these two factors results in the energy efficiency of M-PSM peaking under medium traffic rates.

In order to have an in-depth understanding of the above tradeoff, we select an experimental run for each traffic rate in the mobile scenario, and compare their rate usage and sleep interval distributions in Fig. 11. Here, *sleep intervals* refer to the continuous time intervals during which the station stays in the doze state. Since BMI = 1 in the default PSM scheme, all the sleep intervals in PSM are around 100 ms or less, while with M-PSM, the station could exploit sleep and/or nap opportunities and longer sleep intervals are often observed. As shown in Fig. 11(b), when the traffic rate is low (i.e., 0.01 Mbps), most of the sleep intervals are longer than 100 ms with M-PSM. In fact, more than 80% of the sleep intervals are

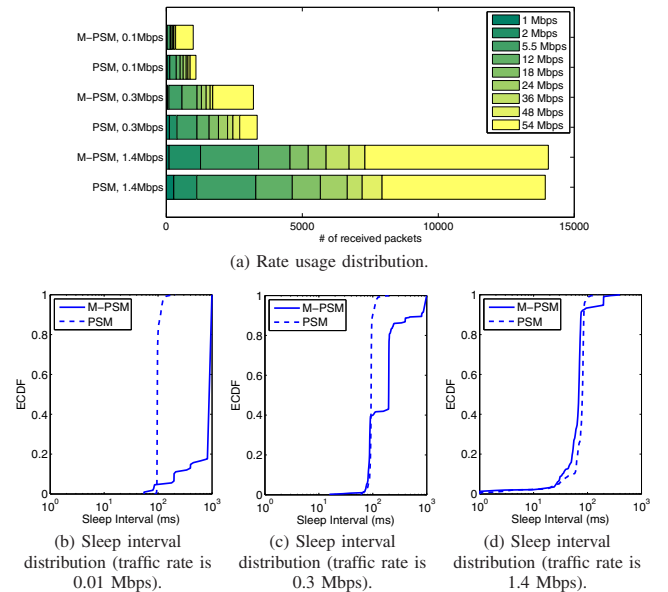


Fig. 11. Comparison of rate usage and sleep interval distributions of three selected experimental runs in the mobile scenario.

close to 1000 ms. This means that the station is able to identify the available sleep opportunities during the station’s movement towards the AP. As a result, almost all the packets that arrive during the station’s movement towards the AP are buffered to be transmitted later at the highest rate of 54 Mbps, as shown in the first horizontal bar in Fig. 11(a). In contrast, PSM blindly wakes up to listen for every single beacon announcement and retrieves the packets whenever the queue is nonempty. Therefore, its rate usage is more evenly distributed. On the other hand, when the traffic rate is high (i.e., 1.4 Mbps), very few sleep opportunities are available to the station and M-PSM performs similarly to the PSM, evidenced by the similar rate usage and sleep interval distributions between them.

3) *Static Scenario*: In the static scenario, the station stays at a fixed location to communicate with the AP, as shown in Fig. 9. Sleep/active time distributions and energy-efficiency performances are compared in Fig. 12.

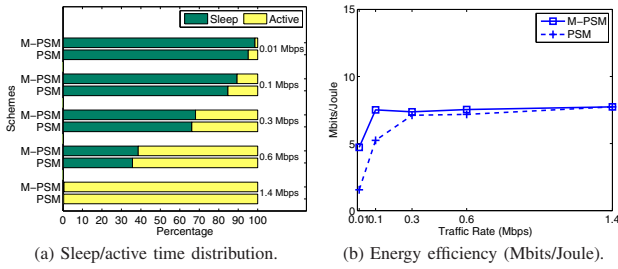


Fig. 12. Comparison of sleep/active time distributions and energy-efficiency performances under different traffic rates. Each point is averaged over five experimental runs.

As expected, the “peaking” phenomenon, which was observed in the mobile scenario and caused by sleep opportunities, does not occur in the static scenario. This is because the channel condition between the static station and the AP is relatively stable; thus sleep opportunities are rarely seen by the station. On the other hand, similar to the mobile scenario, significant performance gain of M-PSM over PSM can be clearly seen under light traffic conditions (i.e., 0.01~0.1 Mbps), as shown in Fig. 12(a). This is due mainly to the nap opportunities identified by M-PSM. This is also confirmed by the significant difference between sleep interval distributions under light traffic conditions, as shown in Figs. 13(a) and (b).

B. Simulation Study

In this section, we perform trace-based simulations using Matlab to further evaluate the performance of M-PSM. Fig. 14(a) shows the SNR trace of one particular experimental run in the mobile scenario, based on which we simulate and compare the performances of M-PSM and PSM under the traffic rate of 0.1 Mbps. An SNR-to-transmission rate mapping table, which is obtained from experimental trials [24] and is consistent with [25], is used in the simulations to determine the proper transmission rates.

In Fig. 14(b), we plot the cumulative energy consumption over time. We can see that the energy consumption of M-PSM first increases in a slower pace than PSM since M-PSM

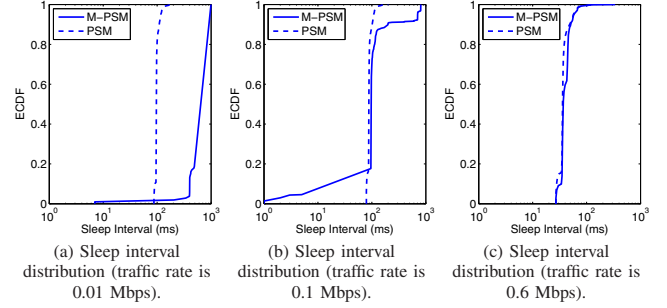


Fig. 13. Comparison of sleep interval distributions of three selected experimental runs in the static scenario.

is able to identify many sleep opportunities when the mobile station is moving towards the AP. Around the 150-second mark, there is a leap in the energy consumption for M-PSM. This is due to the fact that the channel condition becomes very good (higher than 33 dB, as shown in Fig. 14(a)) around that moment, when the station immediately starts issuing PS-Polls to retrieve the buffered packets. The behavior can also be observed from Fig. 14(c), where the queue length of M-PSM drops sharply around the 150-second mark because most of the buffered packets are delivered in a short period of time. After the 160-second mark, the mobile station moves away from the AP. Since then, very few sleep opportunities could be identified. Instead, M-PSM observes a few nap opportunities, which are determined mainly by the traffic condition rather than the mobility pattern. As a result, the station naps for a while (the queue length increases) and then wakes up to retrieve the buffered packets (the queue length decreases), yielding a sawtooth pattern of the queue length variation, as shown in Fig. 14(c).

Comparing with M-PSM, PSM requires a station to wake up every beacon interval to retrieve the buffered packets. Hence, the queue length of PSM is always small (never exceeding

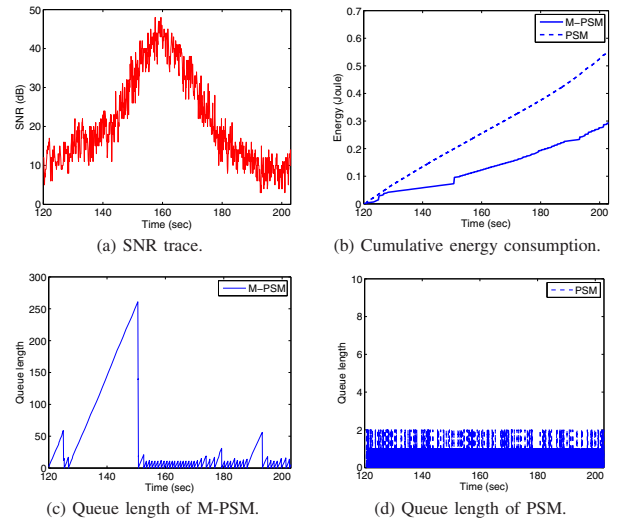


Fig. 14. Comparison of cumulative energy consumption and queue length for M-PSM and PSM when the traffic rate is 0.1 Mbps.

two as depicted in Fig. 14(d)). Since PSM does not recognize or adjust to either mobility pattern or traffic condition, it consumes almost twice amount of energy as M-PSM.

C. Summary

To summarize, we have the following observations:

- M-PSM inherits from PSM. Similar to PSM, M-PSM is best suitable for scenarios when the station is lightly-loaded with delay-insensitive traffic.
- When the traffic rate is low, M-PSM yields a much higher energy efficiency than PSM. This is due to M-PSM's strong ability to detect buffering opportunities and take proper actions upon detection.
- When the traffic rate is high, the performance improvement of M-PSM over PSM is limited, because there exist less buffering opportunities when more traffic is destined for the station. In fact, when the traffic rate is too high, the energy-efficiency performances of both M-PSM and PSM converge to that of the Active Mode. This means that, neither M-PSM nor PSM should be used under heavy traffic conditions, which conforms to the original design goal of PSM (and hence M-PSM) that is to conserve energy for stations under light traffic conditions.
- M-PSM considers the user mobility pattern when making the buffering decisions. Hence, it may save even more energy when the user mobility pattern favors data buffering.
- M-PSM reduces the power consumption of a station via exploiting various buffering opportunities. As a side effect, the packet transmission delay increases inevitably. Therefore, M-PSM is more applicable when the station runs delay-insensitive applications, e.g., file transfer or email. When the station runs delay-sensitive applications, e.g., Skype calls or multimedia streaming, special treatment may be needed if the station is to operate under the proposed M-PSM or the default PSM.

VI. CONCLUSIONS

From experiments with the default 802.11 PSM, we find that (i) more energy may be saved for an 802.11 station by buffering packets at the AP under bad channel conditions and retrieve them later when the channel condition improves; and (ii) beacon reception by an 802.11 station incurs a non-negligible wakeup overhead. Motivated by these observations, we propose a mobility-aware power save mode, called M-PSM, which employs a smart buffering strategy and an adaptive sleep interval adjustment algorithm. We have implemented M-PSM in the Madwifi device driver. Experimental and simulation results show that M-PSM outperforms the default PSM scheme under both mobile and static scenarios, particularly when the traffic is light. Future work includes more rigorous studies of the effects of the power save mode on the delay performance of data delivery services and the extension of M-PSM with handoff considerations for mobile stations that roam between multiple APs.

ACKNOWLEDGMENT

The research reported in this paper was supported in part by the Information Infrastructure Institute (iCube) of Iowa State University and the National Science Foundation (NSF) under Grant CNS 0831874.

REFERENCES

- [1] "IEEE 802.11-2007, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, (Revision of IEEE Std 802.11-1999)," June 2006.
- [2] E. Shih, P. Bahl, and M. J. Sinclair, "Wake on Wireless: An Event Driven Energy Saving Strategy for Battery Operated Devices," in *Proc. ACM MobiCom'02*, September 2002.
- [3] R. Krashinsky and H. Balakrishnan, "Mimimizing Energy for Wireless Web Access with Bounded Slowdown," in *Proc. ACM MobiCom'02*, September 2002.
- [4] D. Qiao and K. G. Shin, "Smart Power-Saving Mode for IEEE 802.11 Wireless LANs," in *Proc. IEEE Infocom'05*, March 2005.
- [5] S. Jin, K. Han, and S. Choi, "A Novel Idle Mode Operation in IEEE 802.11 WLANs," in *Proc. IEEE ICC'06*, June 2006.
- [6] —, "A Novel Idle Mode Operation in IEEE 802.11 WLANs: Prototype Implementation and Empirical Evaluation," in *Proc. ACM WMASH'06*, September 2006.
- [7] —, "Idle Mode for Deep Power Save in IEEE 802.11 WLANs," *Journal of Communications and Networks*, vol. 12, no. 5, pp. 480–491, October 2010.
- [8] Y. He and R. Yuan, "A Novel Scheduled Power Saving Mechanism for 802.11 Wireless LANs," *IEEE Transactions on Mobile Computing*, vol. 8, pp. 1368–1383, October 2009.
- [9] Y. He, R. Yuan, X. Ma, and J. Li, "Analysis of the Impact of Background Traffic on the Performance of 802.11 Power Saving Mechanism," *IEEE Communications Letters*, vol. 13, no. 3, pp. 164–166, March 2009.
- [10] E. Rozner, V. Navda, R. Ramjee, and S. Rayanchu, "NAPman: Network-Assisted Power Management for WiFi Devices," in *Proc. ACM MobiSys'10*, June 2010.
- [11] J. Liu and L. Zhong, "Micro Power Management of Active 802.11 Interfaces," in *Proc. ACM MobiSys'08*, June 2008.
- [12] Y. Chen, N. Smavatkul, and S. Emeott, "Power Management for VoIP over IEEE 802.11 WLAN," in *IEEE WCNC'04*, March 2004.
- [13] B. Gleeson, D. Picovici, R. Skehill, and J. Nelson, "Exploring Power Saving in 802.11 VoIP Wireless Links," in *Proc. ACM IWCMC'06*, July 2006.
- [14] Y. Agarwal, R. Chandra, A. Wolman, V. Bahl, K. Chin, and R. Gupta, "Wireless Wakeups Revisited: Energy Management for Voip over Wi-Fi Smartphones," in *Proc. ACM MobiSys'07*, June 2007.
- [15] V. Nambodiri and L. Gao, "Energy-Efficient VoIP over Wireless LANs," *IEEE Transactions on Mobile Computing*, vol. 9, no. 4, pp. 566–581, April 2009.
- [16] F. R. Dogar, P. Steenkiste, and K. Papagiannaki, "Catnap: Exploiting High Bandwidth Wireless Interfaces to Save Energy for Mobile Devices," in *Proc. ACM MobiSys'10*, June 2010.
- [17] M. C. Rosu, C. M. Olsen, C. Narayanaswami, and L. Luo, "PAWP: A Power Aware Web Proxy for Wireless Lan Clients," in *Proc. IEEE WMCSA'04*, December 2004.
- [18] G. Anastasi, M. Conti, E. Gregori, and A. Passarella, "802.11 Power-Saving Mode for Mobile Computing in Wi-Fi Hotspots: Limitations, Enhancements and Open Issues," *Wireless Networks*, vol. 14, no. 6, pp. 745–768, December 2008.
- [19] A. Schulman, V. Navda, R. Ramjee, N. Spring, P. Deshpande, C. Grunewald, K. Jain, and V. N. Padmanabhan, "Bartendr: A Practical Approach to Energy-aware Cellular Data Scheduling," in *Proc. ACM MobiCom'10*, September 2010.
- [20] Multiband Atheros Driver for Wifi, <http://www.madwifi.org/>.
- [21] X. Chen, P. Gangwal, and D. Qiao, "Practical Rate Adaptation in Mobile Environments," in *Proc. IEEE PerCom'09*, March 2009.
- [22] Iperf, <http://dast.nlanr.net/projects/Iperf>.
- [23] Wi-Fi power management: Road warriors beware, <http://blogs.techrepublic.com/networking/?p=533>, May 2008.
- [24] X. Chen and D. Qiao, "HaND: Fast Handoff with Null Dwell Time for IEEE 802.11 Networks," in *Proc. IEEE Infocom'10*, March 2010.
- [25] J. Zhang, K. Tan, J. Zhao, H. Wu, and Y. Zhang, "A Practical SNR-Guided Rate Adaptation," in *Proc. IEEE Infocom'08*, April 2008.