Centralized PSM: An AP-centric Power Saving Mode for 802.11 Infrastructure Networks

Yi Xie
Department of Computer Science
Xiamen University, China
Email: csyxie@xmu.edu.cn

Xiapu Luo
Department of Computing
The Hong Kong Polytechnic University
Email: csxluo@comp.polyu.edu.hk

Rocky K. C. Chang
Department of Computing
The Hong Kong Polytechnic University
Email: csrchang@comp.polyu.edu.hk

***Abstract*—Energy management in a wireless LAN is an important problem, as the viability of wireless devices depends very much on their battery life. In this paper, we propose a *centralized power saving mode* (C-PSM), an AP-centric PSM for 802.11 infrastructure networks. Having the AP select optimal PSM parameters, such as the beacon and listen intervals, C-PSM is able to maximize the total energy efficiency for all clients. Moreover, C-PSM provides a first-wake-up schedule to further increase the energy efficiency by reducing clients' simultaneous wake-ups. Extensive simulation experiments show that C-PSM outperforms the standard PSM by a very significant margin. In our set of experiments, C-PSM reduces power consumption and increases energy efficiency by as much as 76% and 320%, respectively. As a side benefit, C-PSM also decreases the frame buffering delay at the AP by 88%. The wake-up schedule can save clients' energy consumption by 22% at most. Moreover, the improvement increases with the number of clients.**

I. INTRODUCTION

As more and more applications can be run on mobile and wireless platforms, the problem of improving wireless clients' energy efficiency is becoming very important. Since the wireless communication component is one of the major sources of energy consumption, several approaches have been proposed to alleviate the problem, including power saving mode (PSM) that puts an idle client into a low-power mode, transmission power control, packet transmission scheduling, and cross-layer methods.

The PSM in IEEE 802.11 allows a client to sleep whenever it is idle. The clients wake up periodically to check whether they have frames buffered at the access point (AP) via AP's beacon frames. The AP broadcasts the beacon frames every *beacon interval* (BI); each client wakes up every *listen interval* (LI). Both BI and LI are configurable, and their settings directly influence the PSM's performance. However, the standard does not prescribe how the BI and LI should be configured; therefore, default values are often used. Not only PSM cannot adapt to the traffic and configuration dynamics inherent in typical wireless networks, the PSM is also reported to have adverse impact on application performance, such as short TCP connections [6].

To address these shortcomings, many new power-saving schemes have been proposed. A class of them [9], [11], [6] enables each client to save energy by reducing the number of unnecessary wake-ups, for example, designing an optimal wake-up schedule. These user-centric schemes, however, do not reduce energy consumption resulted from channel contention which, as we will show, is a major source of energy wastage. Another class is based on an AP-centric approach which relies on the AP to improve the energy efficiency of all clients in the network. The AP could adopt a packet transmission schedule (e.g., [5], [7]) or determine the clients' wake-up schedules (e.g., [8]) to reduce channel contentions.

We propose C-PSM, a new AP-centric PSM, to let the AP choose the best BI and LIs for all clients based on the their traffic patterns. These intervals are chosen to reduce energy consumption caused by unnecessary wake-ups and channel contentions. The energy wasted in channel contentions could be very significant, because all the clients involved cannot go to sleep during the contention period. To further reduce energy consumption, C-PSM includes a first-wake-up schedule to reduce clients' simultaneous wake-ups. Moreover, the AP assigns optimal minimum congestion windows to the clients, so that a client that wakes up less frequently will be able to retransmit earlier.

There are a few important differences between C-PSM and other AP-centric schemes. First, C-PSM conforms to the PSM standard, whereas other AP-centric schemes, such as [5], do not. The only additional mechanism required for C-PSM is to notify the clients of their parameter settings which could be accomplished through the beacon transmission channel. Second, C-PSM does not rely on computation-expensive packet scheduling which is required in [5], [8], [7]. Instead, the AP in C-PSM simply observes the statistics of the packet arrival patterns. Third, C-PSM is designed independent of the upper-layer protocols. Therefore, it could be used for any mix of network protocols. However, other AP-centric schemes, such as [9], are designed only for HTTP clients.

We structure the rest of this paper as follows. Section II first summarizes energy-saving schemes previously proposed for IEEE 802.11 infrastructure networks. Section III then describes the PSM, the system model, and the simulator used in this paper. To motivate the design of C-PSM, section IV analyzes the impacts of BI and LIs on energy efficiency. Section V presents C-PSM, and section VI evaluates the performance of C-PSM based on simulation experiments. Section VII summarizes this paper with future work.

This work was supported by a grant from the Research Grant Council of the Hong Kong SAR, China (Project No. PolyU 5146/01E). It was done when Yi Xie was with The Hong Kong Polytechnic University.

II. Related work

User-centric approaches have been proposed to improve PSM's performance. Nath et al. [9], for example, proposed a dynamic wake-up period in which each client chooses its LI according to the current round-trip time of its TCP connection. The Bounded Slowdown Protocol [6], another user-centric method, allows a client to increase its LI when the period of idleness prolongs. In Smart PSM [11], each client determines whether it will enter into the PSM, depending on their traffic condition. Although the user-centric methods are effective in reducing *individual* client's energy consumption, they do not address the power consumption due to channel contention. Moreover, it is not clear whether these schemes remain effective when some clients do not employ them.

AP-centric approaches let the AP control the PSM operations. Almost all AP-centric schemes support multiple clients and attempt to eliminate channel contention. The wake-up schedule proposed in [8], for example, redesigns the TIM to let only one client retrieve its buffered frame. The AP in the scheduled PSM [5] assigns slices of a BI for the clients' buffered frames. The scheme proposed in [7] computes an optimal BI and designs an energy-efficient scheduler for frame transmissions within one BI. Although these schemes generally perform well, the cost is a computation-intensive scheduling algorithm for the wake-up and frame transmissions.

III. Models and notations

A. The IEEE 802.11 PSM

The PSM allows a wireless client to sleep instead of staying in active state all the time and the AP to buffer the frames for them. Let the AP's BI be β milliseconds (ms). The periodic beacon frame includes a *Traffic Indication Map* (TIM) that indicates whether the AP has buffered frames for each PSM-enabled client. On the other hand, each PSM-enabled client's LI is a multiple of BI; therefore, the BI actually determines the LI's granularity. Let the value of LI be $\gamma \times \beta$ milliseconds, where $\gamma \geq 1$. For the PSM, the default settings are $\beta = 100$ ms and $\gamma = 1$.

Figure 1 [4] illustrates the PSM operation for two clients s_1 and s_2. The first (second) bit in the TIM indicates the buffer status for client s_1 (s_2). The shaded region shows that the client is in the active state. After waking up for the first time, s_1 is notified of its frames being buffered at the AP through the TIM. The client then sends a PS-Poll frame to retrieve the first frame. If the More Data bit in the received frame is not set, it will return to sleep; otherwise, it will send another PS-Poll frame. The same data exchange repeats until all buffered frames are sent and then the client goes to sleep. If a client wants to send data, it may wake up any time to transmit them.

B. System model

Our model consists of an AP and c PSM-enabled wireless clients s_j, $j = 1, \ldots, c$, which run on IEEE 802.11b with a transmission rate of 11 Mps. The model considers only downlink traffic (i.e., those from the AP to clients). Each traffic source will send frames to a corresponding wireless client. There are two random variables in our models: inter-frame arrival time and the frame size. Let T_j be the inter-frame arrival time for s_j's traffic source and the mean of T_j be δ_j. Let the vector of the client's mean inter-frame arrival times be $\Delta = [\delta_1, \ldots, \delta_c]$. Our current study allows T_j to take on four types of distributions: deterministic (DET), uniform (UNI), exponential (EXP), and Pareto (PAR). On the other hand, the frame size distribution is either deterministic and uniform. Besides β, the AP in C-PSM can also configure the LIs for all clients ($\Gamma = [\gamma_1, \ldots, \gamma_c]$) and the minimal congestion windows for all clients ($\Theta = [\theta_1, \ldots, \theta_c]$).

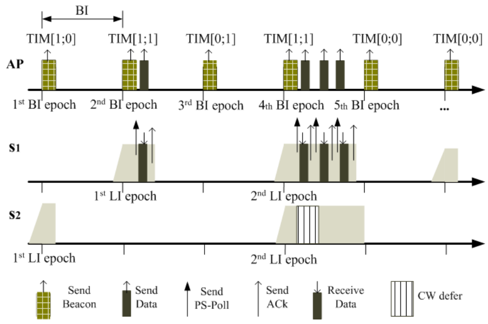


Fig. 1. An example of the PSM operation for two wireless clients.

C. MATLAB-based simulator

Publicly available simulators, like J-SIM [1], have not fully implemented PSM for infrastructure network, such as the beacon and PS-Poll frames. The PSM module [6] in ns-2 [2] only supports a single client. This prompted us to design and implement a new simulator based on MATLAB, which simulates the details of the IEEE 802.11b DCF with PSM, including the PS-Poll, beacon frames (with TIM), backoff algorithm, and congestion window. We have excluded other nonessential elements (e.g., authentication and (de)association) and the RTS/CTS mechanism which is often turned off to increase performance. We have used the simulation parameters in Table I for the experiments conducted in this paper[1]. The set of power and energy consumption for Lucent IEEE 802.11 Wavelan PC card has been used widely and was evaluated [3].

TABLE I
SIMULATION PARAMETERS USED IN THIS PAPER.

Parameters	Values	Parameters	Values
Number of clients	1 to 20	slotTime	20 μs
Data transmission rate (DTR)	11 Mbps	SIFS	10 μs
Basic transmission rate (BTR)	2 Mbps	DIFS	50 μs
Data frame size (DFS)	512 bytes	Transmission power	1.4 W
Beacon frame size	28 bytes	Reception power	0.9 W
PS-Poll frame size (PFS)	14 bytes	Idle power	0.7 W
ACK frame size (AFS)	14 bytes	Sleeping power	0.060 W
		Wake-up energy	0.003 J

The simulator produces detailed trace files which record frame exchanges, channel collisions, and clients' mode transitions. By carefully analyzing the trace files, we have validated the correctness of the PSM simulator. Moreover, we have obtained the following metrics from the trace files:

1) P: the total power consumed by the clients in watt (W).

[1]The simulator uses only the long preamble (192μs). The transmission time of each frame is therefore equal to $192 + \frac{frame\ size \times 8}{transmission\ rate} \mu s$.

2) T: the total client throughput in bits per second (bps).
3) $R_{T/P}$: $\frac{T}{P}$, the total energy efficiency metric in bits per joule (bpJ).
4) $R_{c/t}$: $\frac{N_c}{N_t}$, where N_t is the total number of transmission attempts by all clients, and N_c is the total number of channel collisions experienced by all clients.
5) $R_{u/w}$: $\frac{N_u}{N_w}$, where N_w is the total number of wake-ups by all clients, and N_u is the total number of unnecessary wake-ups by all clients.
6) $R_{bB/B,k}$: $\frac{N_{bB,k}}{N_B}$, where N_B is the total number of BIs, and $N_{bB,k}$ is the total number of BIs in which k clients are involved in channel contention, $k \geq 2$.
7) d_j: the average buffering delay of s_j's frames at the AP.

After observing the time of convergence from several preliminary experiments, we have let each experiment run for at least 20 seconds in simulation time. We have also repeated each simulation for 20 times and reported their average values. All the results reported here fall within a 95% confidence level.

IV. A PRELIMINARY ANALYSIS

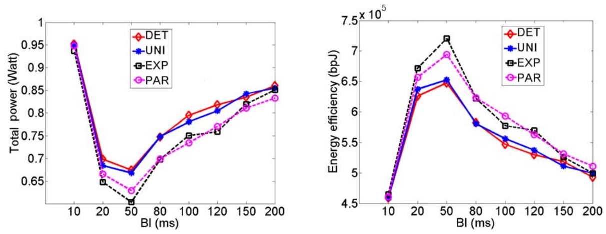
There are two main sources of energy wastage: unnecessary wake-ups and channel contention. Clearly, the individual LI has a direct impact on the number of unnecessary wake-ups. Overly-frequent wake-ups will consume a significant amount of energy. For example, s_2 in Figure 1 wakes up the first time to find no frames buffered for it.

Energy wastage due to channel contention, however, is more complicated. Back to Figure 1 again, s_2 wakes up the second time to find frames buffered at the AP. It has lost to s_1 after contending for the channel during the PS-Poll transmissions. s_2 then stays in the active mode during the backoff process which could take a long time. Therefore, rescheduling the active clients' wake-ups to different times will help reduce this source of consumptions (i.e., nonoverlapping of LIs). Note that reducing the BI value will also help, because, as mentioned before, the BI value determines the LI's granularity.

A. Evaluating the impacts of beacon and listen intervals

Besides the parameters given in Table I, we have chosen Δ, such that the network is lightly loaded. It has been shown recently that the sleeping mechanism is valuable when the network utilization (ρ) is less than 30% [10]. Therefore, we consider only those Δs that satisfy $\rho = b_{min} \sum_{j=1}^{c} 1/\delta_j < 30\%$, where b_{min} is the minimum transmission time for one data frame (i.e., without using PSM or suffering from channel contention). Note that $b_{min} = \frac{DFS}{DTR} + \frac{PFS+AFS}{BTR} + DIFS + 2SIFS \approx 1.13ms$. The results presented here are based on $c = 2$ and $\Delta = [15;25]$ ms, where $\rho \approx 12\% < 30\%$.

Impact of BI: We have studied the impact of β for the range of 10∼200 ms with the default PSM settings ($\Gamma = [1;1]$ and $\Theta = [31;31]$). Shown in Figure 2(a), P is high when β is too small, because much energy is wasted on clients' frequent wake-ups. When β is too large, many frames have been accumulated at the AP. Therefore, energy is wasted on channel contention. The optimal β in this example is 50ms, instead of the default value of 100 ms. Moreover, since $R_{T/P}$ is inversely proportional to P, Figure 2(b) behaves similar to Figure 2(a).



(a) P verses β, $\gamma_1 = \gamma_2 = 1$ (b) $R_{T/P}$ verses β, $\gamma_1 = \gamma_2 = 1$

Fig. 2. Impact of BI for $c = 2$ and $\Delta = [15;25]$ ms.

Impact of LIs: The choice of Γ can also influence the clients' energy consumption and communication performance. In our simulation study, we have fixed β to 50 ms and considered different Γs for the two clients. Table II for the EXP distribution clearly shows that the default case of $[1;1]$ is not optimal for the six performance metrics. The best value for each metric is underlined. Overall, the case of $[1;2]$ achieves the best performance. Note that $\frac{\gamma_1}{\gamma_2}$ is closest to $\frac{\delta_1}{\delta_2}$ for $\Gamma = [1;2]$. We have observed similar results for the other three inter-frame arrival time distributions.

TABLE II
SIMULATION RESULTS FOR DIFFERENT ΓS UNDER THE EXP INTER-FRAME ARRIVAL DISTRIBUTION FOR $c = 2$, $\beta = 50$ MS, AND $\Delta = [15;25]$ MS.

Γ	$R_{c/t}$	$R_{u/w}$	$R_{bB/B,2}$	P(W)	$R_{T/P}$(ms)	d_1(ms)	d_2
$[1;1]$	1.54%	11.51%	81.37%	0.6109	7.1578	37.4	32.3
$[1;2]$	1.04%	4.97%	42.32%	0.5487	7.9674	29.8	60.0
$[2;1]$	1.07%	12.24%	46.79%	0.6032	7.2316	81.0	28.0
$[2;2]$	1.25%	1.67%	49.16%	0.7470	5.8260	125.4	61.3

V. CENTRALIZED PSM

In C-PSM, the AP determines optimal PSM settings for itself and all clients. It first searches for optimal β (denoted by β^*) and optimal Γ (denoted by Γ^*) based on the frame arrival patterns. The intervals are selected to reduce the number of unnecessary wake-ups and channel contention. The AP also obtains a Θ (denoted by Θ^*) to ensure that any client will not be denied channel access for too long.

The inputs to the AP's algorithm include Δ, β_{min}, ε_β, and ε_Θ. β_{min} is a lower bound of β^*, and ε_β and ε_Θ are the step sizes for searching β^* and Θ^*, respectively. The main algorithm executes three steps to yield β^*, Γ^*, and Θ^*.

Step 1 (Determining the β^* and Γ^* candidates) obtains a number of β^* and Γ^* candidates for the second step.

We first consider a Γ^* candidate: $[L_1;\ldots;L_c]$. We further let $L_j = \alpha_j \times \delta_j$, where $\alpha_j \geq 1$ is an integer scaling factor. To reduce unnecessary wake-up, the probability that an awaken client finds an empty buffer at the AP (denoted by Pr_0) should be less than a given threshold $0 < \xi \leq 1$. The choice of this threshold reflects the tradeoff between the number of unnecessary wake-ups and the period of channel contention. If ξ is too large, the LI may be short, thus inducing a lot of unnecessary wake-ups. If ξ is too small, the frames buffered during the long LI may cause channel contention. After running a number of empirical simulations, we have selected $\xi = 0.05$. Then, α_j is the smallest integer which satisfies $Pr_0 \le 0.05$. Table III shows examples of determining α_j under the four inter-frame arrival time distributions.

To determine the β^* candidates, following the guideline in [9], we set β_{min} to 10 ms. To determine the upper bound of β, we note that BI should not be larger than any client's LI. Therefore, the upper bound of β is given by $\min_{\forall j} L_j$. We then select $n+1$ BI candidates uniformly within the range of $[\beta_{min}, \min_{\forall j} L_j]$, where $n = \lfloor (\min_{\forall j} L_j - \beta_{min})/\varepsilon_\beta \rfloor$. Moreover, for each β^* candidate, we consider three Γ^* candidates:

1) $\Gamma_{i,1} = [\lceil L_1/\beta_i \rceil; \cdots; \lceil L_c/\beta_i \rceil]$,
2) $\Gamma_{i,1} = [\langle L_1/\beta_i \rangle; \cdots; \langle L_c/\beta_i \rangle]$, where $\langle x \rangle$ gives the round-off value of a real number x, and
3) $\Gamma_{i,1} = [\lfloor L_1/\beta_i \rfloor; \cdots; \lfloor L_c/\beta_i \rfloor]$.

TABLE III
Pr_0 VS. α UNDER DIFFERENT TRAFFIC DISTRIBUTIONS.

Distribution	Pr_0				
	$\alpha = 1$	$\alpha = 2$	$\alpha = \mathbf{3}$	$\alpha = 4$	$\alpha = 5$
DET	0	0	0	0	0
UNI	0.5	0	0	0	0
EXP	0.3679	0.1353	**0.0498**	0.0183	0.0067
PAR (k=1/3)	0.2963	0.0787	0.0315	0.0156	0.0089

Step 2 (Determining β^* and Γ^*) obtains the best β^* and Γ^* candidates from the set identified from step 1. The criterion is based on minimizing the number of simultaneous wake-ups. There are two sub-steps to achieving the goal.

In the first sub-step, we search for the best Γ for each β^* candidate obtained in step 1. That is, for a given β_i obtained in step 1, we select the best Γ from $\Gamma_{i,1}$, $\Gamma_{i,2}$, and $\Gamma_{i,3}$ that minimizes the number of simultaneous wake-ups. Since γ_js are integer valued, we choose the best Γ based on the largest LCM and denote this choice by Γ_i^*. Note that the LCM gives the minimal number of BIs for which two or more clients wake up simultaneously. Therefore, a larger LCM means a lower number of simultaneous wake-ups.

In the second sub-step, given (β_i, Γ_i^*), $i = 1, \ldots, n+1$, we select the best Γ from the Γ_i^*s that will again minimize simultaneous wake-up. The criterion is based on the largest spread of their elements from one another which is measured by the ratio of the standard deviation and the mean of the elements in Γ_i^*. Therefore, Γ^* is given by the Γ_i^* that gives the highest ratio, and β^* is the corresponding β_i.

Step 3 (Determining Θ^*) seeks to determine Θ^* based on the Γ^* obtained in Step 2. The motivation is to mitigate possible unfairness in the frame buffering delay experienced by the clients. The approach is to assign a smaller θ_j^* to the client with a larger γ_j^*. In this way, the client that wakes up less frequently will have a higher priority to retrieve its frames during channel contention. To do so, we use $\theta_j^* = 31 + \varepsilon_\Theta(\max_{\forall j}(\gamma_j^*) - \gamma_j^*)$. Therefore, the client with the largest γ_j is assigned with the default value ($\theta_j^* = 31$) and others with smaller γ_js will backoff beyond the default.

Besides the main algorithm, the AP in C-PSM may also obtain optimal first-wake-up times for the clients, denoted as $\mathbf{r}^*$, where r_j is s_j's first-wake-up time which is measured in the unit of BI. This optimal *first-wake-up schedule* (FWS) can further decrease the number of simultaneous wake-ups if two or more elements of Γ^* are the same or have the same common factor. Given Γ^*, we can obtain $\mathbf{r}^*$ using a similar stepwise solving method proposed for another wake-up scheduling problem considered in [8]. Due to the limited space, we do not describe the algorithm but will present the results in the next section.

VI. Performance Evaluation

We use the following four performance metrics (power saving, throughput, energy efficiency, and frame buffering delay) to evaluate the performance of the four schemes. S-PSM's quantities are labeled by superscript S, whereas the other three's do not have the superscript. Note that we use S-PSM's performance as a baseline in the metric definitions.

$$
\begin{aligned}
\eta_P &= (P^S - P)/P^S \times 100\%, \\
\eta_T &= (T - T^S)/T^S \times 100\%, \\
\eta_{T/P} &= (R_{T/P} - R^S_{T/P})/R^S_{T/P} \times 100\%, \\
\eta_D &= (\sum_{j=1,\ldots,c} (d_j^S - d_j)/d_j^S)/c \times 100\%.
\end{aligned}
$$

A. Two clients

We first evaluate C-PSM in a two-client system. To evaluate the effectiveness of different components of C-PSM, we consider three different versions. The first one is a "full version" which includes the FWS discussed in the last section. The other two (Scheme-1 and Scheme-2), however, exclude the FWS (i.e., $r_j = 0, \forall j$) and adopt the default congestion window size (i.e., $\theta_j = 31, \forall j$). The difference between Scheme-1 and Scheme-2 is that Scheme-2 adopts the default Γ value ($\gamma_j = 1, \forall j$). We also consider the standard PSM (denoted by S-PSM) in the evaluation. To sum up:

1) C-PSM : β^*, Γ^*, Θ^*, and $\mathbf{r}^*$;
2) Scheme-1: β^*, Γ^*, $\theta_j = 31$, and $r_j = 0, \forall j$;
3) Scheme-2: β^*, $\gamma_j = 1$, $\theta_j = 31$ and $r_j = 0, \forall j$;
4) S-PSM : $\beta = 100$ ms, $\gamma_j = 1$, $\theta_j = 31$ and $r_j = 0, \forall j$.

Given $\Delta = [15; 25]$ ms, Table IV shows the optimal PSM parameters obtained by an AP in C-PSM under different traffic distributions. Note that $r_j^* = 0, \forall i$, for this scenario; therefore, C-PSM and Scheme-1 differ only in the adoption of Θ^*.

As shown in Table V, C-PSM's improvement over S-PSM can be very significant. For example, C-PSM reduces power consumption by 29.37%, improves energy efficiency by 43.01%, and reduces average buffering delay by 54.8% under the EXP traffic distribution. C-PSM's η_P and $\eta_{T/P}$ are higher than Scheme-1's and Scheme-2's. Therefore, employing β^*, Γ^* and Θ^* gives the best performance in power saving and energy efficiency.

TABLE IV
Optimal parameters of C-PSM ($\varepsilon_\beta = 2$ ms and $\varepsilon_\Theta = 8$).

Δ (ms)	distribution	β^* (ms)	Γ^*	Θ^*	$\mathbf{r}^*$
[15;25]	DET	10	[2;3]	[39;31]	[0;0]
	UNI	26	[1;2]	[39;31]	[0;0]
	EXP,PAR	38	[1;2]	[39;31]	[0;0]
[20;30;30]	DET	16	[1;2;2]	[39;31;31]	[0;0;1]
	UNI	30	[1;2;2]	[39;31;31]	[0;0;1]
	EXP,PAR	46	[1;2;2]	[39;31;31]	[0;0;1]

The results in Table V also show that the performance benefit of adopting both β^* and Γ^* is significant, because

TABLE V
COMPARING C-PSM, SCHEME-1 AND SCHEME-2 WITH $\Delta = [15;25]$ MS.

index, %	scheme	DET	UNI	EXP	PAR
η_P	C-PSM	25.41	28.75	29.73	28.13
	Scheme-1	24.91	27.28	27.53	26.47
	Scheme-2	-20.82	17.52	21.10	16.48
$\eta_{T/P}$	C-PSM	34.63	41.18	43.01	39.76
	Scheme-1	33.71	38.33	38.65	36.57
	Scheme-2	-16.88	21.95	27.38	20.30
η_D	C-PSM	82.33	68.79	54.80	54.18
	Scheme-1	82.08	68.15	53.07	53.56
	Scheme-2	94.54	79.79	69.88	68.75
η_T	C-PSM	0.41	0.59	0.50	0.45
	Scheme-1	0.40	0.60	0.48	0.42
	Scheme-2	0.43	0.58	0.50	0.47

Scheme-1 outperforms Scheme-2 by a large margin in η_P, $\eta_{T/P}$, and η_D. Without using Γ^*, Scheme-2's performance is even worse than S-PSM for a small number of cases. Additionally, the performance improvement due to Θ^* is minor. For most cases, C-PSM outperforms Scheme-1 by only a small percentage (the largest deviation is in $\eta_{T/P}$ for EXP).

B. More than two clients

We have also carried out experiments for more than two clients. For three clients, we have set two of the clients' mean inter-frame arrival times to be the same. For $\Delta = [20;30;30]$ ($\rho = 13.18\% < 30\%$), C-PSM's optimal parameters are given in Table IV. Note that when the elements of Γ^* are not relatively prime, C-PSM will adopt the FWS. In this case, the FWS uses $\mathbf{r}^* = [0;0;1]$: s_1 and s_2 wake up at the beginning of the simulation, and s_3 defers its first wake-up time by β^*.

We have compared C-PSM, C-PSM without the FWS, and S-PSM. Table VI shows that C-PSM outperforms S-PSM in all performance metrics. C-PSM saves energy by shortening the period of channel contention which is shown by its lowest frame buffering delay. C-PSM saves energy also by reducing channel contentions. It prevents all clients from waking up simultaneously to contend for the channel, thus reducing $R_{bB/B,3}$ to 0% (however, S-PSM's $R_{bB/B,3}$ is 92.29%). Moreover, C-PSM consumes a small amount of energy on unnecessary wake-ups, because its $R_{u/w}$ is only 10%, and it reduces collisions ($R_{c/t}$) by about one third.

TABLE VI
COMPARING C-PSM, C-PSM WITHOUT FWS, AND S-PSM UNDER THE EXP TRAFFIC DISTRIBUTION WITH $\Delta = [20;30;30]$ MS.

Metrics	C-PSM	C-PSM without FWS	S-PSM
P (Watt)	0.8242	0.9591	1.3037
T (10^5 bps)	4.7538	4.7536	4.7257
$R_{T/P}$ (10^5 bpJ)	5.7675	4.9563	3.6248
d_1(ms)	34.4	36.3	234.2
d_2(ms)	55.8	63.2	84.5
d_3(ms)	54.9	64.7	87.4
$R_{c/t}$	1.44%	1.78%	2.14%
$R_{u/w}$	10.06%	10.66%	4.86%
$R_{bB/B,2}$	83.83%	10.47%	7.64%
$R_{bB/B,3}$	**0**	39.05%	92.29%

C-PSM without the FWS still outperforms S-PSM; however, the absence of the FWS degrades C-PSM's performance. Without the FWS, the three awaken clients compete for receiving data in 39.05% of the BIs; using the FWS can totally avoid such situation (i.e., $R_{bB/B,3} = 0\%$). Moreover, although C-PSM without FWS has a much lower $R_{bB/B,2}$ than C-PSM, this advantage cannot offset the disadvantage of the contention involving all three clients. Therefore, we conjecture that the adverse effect of contention increases exponentially with the number of clients involved in the contention. Our experiment results have illustrated the importance of having the FWS to reduce the all-involved contentions.

We have finally designed a set of experiments to evaluate C-PSM's performance when the number of clients is increased up to 20. We let $\delta_j = 10c$ ms, $j = 1, \ldots, c$. The total amount of traffic of the symmetric clients will not change with c, because the total arrival rate of frames $\sum_{j=1}^{c} \frac{1}{\delta_j}$ always equals 100 frames per second, creating a traffic intensity of $\rho = 11.3\% < 30\%$. Table VII shows that C-PSM also achieves significant improvement in a large network. The value of $\eta_{T/P}$ can reach 327.07% for a network of eight clients.

TABLE VII
C-PSM'S PERFORMANCE VERSES THE NUMBER OF CLIENTS UNDER THE EXP TRAFFIC DISTRIBUTION WITH $\delta_j = 10c$ MS.

Index, %	c=2	c=4	c=8	c=12	c=16	c=20
η_P	51.09	70.33	76.07	70.98	72.27	72.14
$\eta_{T/P}$	105.04	238.69	327.07	257.23	277.64	281.02
η_D	64.09	70.89	82.68	85.02	87.04	88.43
η_T	0.29	0.48	2.19	3.68	4.73	6.14

VII. CONCLUSIONS AND FUTURE WORKS

We have proposed the centralized PSM (C-PSM) that increases energy efficiency of all wireless clients in an infrastructure network. The AP in C-PSM computes the best beacon interval, listen intervals, and minimal congestion window sizes according to the clients' traffic patterns. The jointly optimized intervals can reduce unnecessary wake-ups and channel contention which collectively translate into energy saving and reduction in the buffering delay. The optimal minimal congestion windows are also helpful to mitigate unfairness among clients. C-PSM further increases the energy efficiency by scheduling the first-wake-up times to reduce simultaneous wake-ups. Moreover, C-PSM's performance increases with the number of clients. In future work, we will further improve the main algorithm's performance and extend C-PSM to support different traffic models.

REFERENCES

[1] The J-SIM site. http://sites.google.com/site/jsimofficial/.

[2] The network simulator - ns-2. http://www.isi.edu/nsnam/ns/.

[3] L. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *Proc. IEEE INFOCOM*, 2001.

[4] M. Gast. *802.11 Wireless Networks: The Definitive Guide*. O'Reilly Media, Inc., 2 edition, 2005.

[5] Y. He, R. Yuan, X. Ma, J. Li, and C. Wang. Scheduled PSM for minimizing energy in wireless LANs. In *Proc. IEEE ICNP*, 2007.

[6] R. Krashinsky and H. Balakrishnan. Minimizing energy for wireless web access with bounded slowdown. *Wireless Networks*, 11, 2005.

[7] J. Lee, C. Rosenberg, and E. Chong. Energy efficient schedulers in wireless networks: design and optimization. *Mobile Networks and Applications*, 11(3), June 2006.

[8] H. Lin, S. Huang, and R. Jan. A power-saving scheduling for infrastructure-mode 802.11 wireless LANs. *Computer Communications*, 29:3483–3492, 2006.

[9] S. Nath, Z. Anderson, and S. Seshan. Choosing beacon periods to improve response times for wireless HTTP clients. In *Proc. MobiWac*, 2004.

[10] S. Nedevschi, L. Popa, and G. Iannaccone. Reducing networking energy consumption via sleeping and rate-adaptation. In *Proc. USENIX NSDI*, 2008.

[11] D. Qiao and K. Shin. Smart power-saving mode for IEEE 802.11 wireless LANs. In *Proc. IEEE INFOCOM*, 2005.