Enayet et al. EURASIP Journal on Wireless Communications and Networking (2016) 2016:27
DOI 10.1186/s13638-016-0522-7

EURASIP Journal on Wireless Communications and Networking
a SpringerOpen Journal

RESEARCH Open Access

PowerNap: a power-aware distributed Wi-Fi access point scheduling algorithm

Asma Enayet[1†], Nusrat Mehajabin[1†], Md Abdur Razzaque[1†*], Choong Seon Hong[2] and Mohammad Mehedi Hassan[3]

Abstract

In this paper, we design a power-aware distributed access point scheduling algorithm, PowerNap, to enhance power conservation of co-existing multiple access points (APs) each having multiple clients in the same wireless vicinity. This consequently addresses low channel utilization, degraded throughput, and unfairness problems of Wi-Fi networks in an energy-efficient way. PowerNap schedules transmission periods of APs according to their traffic loads to ensure fair access of the medium from their respective clients' perspective. It supports dynamic rescheduling of AP transmission periods to aid client mobility and traffic fluctuations. The scheduling also ensures that no two APs, in a shared environment, wake up their clients at exactly the same time, decreasing data packet collisions and thus increasing network throughput and energy-efficiency. PowerNap achieves decentralization by exploiting single-hop neighborhood information (e.g., traffic loads) only, and thus, it is scalable. Performance evaluations, carried out in ns-3, depict that the effectiveness of the proposed PowerNap algorithm surpasses the state-of-the-art approaches in terms of energy consumption, network throughput, and fairness.

Keywords: Wi-Fi networks, Energy-efficiency, AP scheduling algorithm, Throughput, Smartphone

1 Introduction

Wi-Fi-enabled smart devices run numerous applications which require added processing power, untethered Internet access and battery-life [1]. Uninterrupted services on the fly without compensating the number of running applications [2] demand innovative solutions for battery power conservation. This has always been a critical and highly anticipated goal for the research community to optimize energy consumption of the Wi-Fi clients [3–6]. Achieving the said optimization expectations will result in improved user experience and productivity.

Among all the energy-hungry components (e.g., LCD display, networking sub-system, sensors, and communication and computation circuitries [7]) of a Smartphone, networking sub-systems energy management deserves a special attention as 10–50 % (e.g., 3G, GSM, and Wi-Fi) of total energy consumed is associated with it [8–10]. Recent research for Wi-Fi radio energy management shows a large portion of consumed power is wasted due to network contention at the access points (AP) level [11, 12].

One of the state-of-the-art solutions is Wi-Fi traffic load optimization. And the default Wi-Fi traffic optimization is the IEEE 802.11 power saving mode (PSM) [13]. It relies on the assumption that the network activity will be idle for a predefined duration [14], which is hardly true for present systems. Moreover, the PSM does not support multiple clients or multiple APs in the same vicinity simultaneously [15], resulting in heavy contention in the network. NAPman [11] mitigates the single client problem of the PSM and support multiple clients through AP virtualization. However, virtualization is not same as multiple physical APs, and in such situations, the NAPman suffers a lot. SleepWell [12] proposes an algorithm to schedule multiple APs in a neighborhood so that they do not overlap. However, it fails to detect hidden terminals which consequently add unfairness and low channel utilization.

In this paper, we design a distributed access point scheduling algorithm, PowerNap. The key philosophy of PowerNap is to cut energy losses by scheduling the APs transmissions in a non-overlapping manner and allocate

*Correspondence: razzaque@du.ac.bd
†Equal contributors
[1]Green Networking Research (GNR) Group, Department of Computer Science and Engineering, University of Dhaka, Dhaka, 1000, Bangladesh
Full list of author information is available at the end of the article

transmission periods according to the AP's weighted traffic share. A preliminary version of this work has been published in [16] which is augmented with performance evaluation using NS-3, analysis of undershooting parameter α, fairness level measurement using Jain's fairness index, and protocol operation overhead of PowerNap algorithm. Dynamic rescheduling is also employed to aid robustness against client mobility and traffic fluctuations. The key contributions of this work are summarized below:

- We design an energy-conserving access point (AP) scheduling algorithm, PowerNap, where every AP exploits only single-hop neighborhood traffic information for scheduling decisions, which makes the algorithm a distributed and scalable one.
- A PowerNap-enabled AP computes its weighted fair share with reference to the neighboring (e.g., one hop neighborhood) AP's traffic loads (e.g., weighted fair share is proportional to traffic load) to allocate transmission periods to the clients, and thus, it avoids unfairness.
- The unused portions of transmission rounds, if there is any, are also distributed in a weighted manner to the contending APs to utilize full potential of the channel bandwidth.
- The necessary modifications and added components in PowerNap require changes in APs only; no client side changes are required. Thus, the proposed PowerNap is easily deployable.
- Our performance evaluation, carried out in ns-3, shows that the proposed PowerNap scheduling algorithm saves up to 27 % energy than the state-of-the-art approaches.

The rest of the paper is organized as follows. Section 2 contains the literature study. In Section 3, we describe the network model and assumptions and the key design principle of the proposed algorithm. Section 4 gives insight into the design and operation details of our proposed scheduling algorithm, PowerNap. Section 5 presents the performance evaluation of PowerNap using ns-3 [17]. Finally, we conclude the paper in Section 6.

2 Related work

The energy conservation of Wi-Fi [18] devices has received many proposals over the last few years [2, 19–21]. These existing mechanisms and techniques can be organized into several categories, e.g., decision by monitoring the link status, bandwidth as a decision variable, scanning vicinity using cellular towers for information, and AP scheduling algorithms [22].

The IEEE 802.11 PSM [13] is the default power saving mechanism for IEEE 802.11 APs. The PSM conserves energy by putting the clients to sleep during idle periods. All the data packets destined for the clients are queued at the AP. It configures the clients to doze off while there is no traffic for a predefined duration (typically 100 ms) and wake them up at low power mode on every beacon interval (100 ms) to check for beacon frames destined to this device. If such a packet is found, the client flips to high power transmission mode. The client device stays on this mode for the next 300 ms irrespective of whether it is receiving packets for that long period or not. It goes back to sleep mode again when it has no packets for next 100 ms. The PSM has no provision for multiple clients waking up at exactly the same time to access the channel; it also fails to support multiple APs in the same wireless vicinity, which leads to heavy contention among the APs. One of the major limitations of PSM is the assumption that the network will be idle for a certain time, which is both unrealistic and unfair. As the clients might not find any idle period and remain in high-power transmission mode indefinitely.

The authors of [23] focused on sharing Smartphone's Internet connection with other devices in the neighborhood domain using Wi-Fi. Their proposed scheme reduces the energy consumption of a mobile AP which adaptively adjusts the sleep and wake-up periods based on the bandwidth asymmetric feature of the AP. Further, it combines idle times between packets in order to provide client a prolonged sleeping time and thus conserves energy.

Low energy data packet aggregation scheme (LEDAS) [24] authors observed the characteristics of network packets. They discovered most of the network packets are small in size and to wake up the entire radio spectrum to transmit a small packet poses huge energy wastage. Using this information, authors decided to aggregate the packets in a large burst and send it at once. Therefore, the radio is only invoked when large packets are waiting to be transmitted. A problem arises when time-sensitive packets need to be dealt with separately. For dealing with time-sensitive packets, LEDAS [24] implemented two different queues; one for time-sensitive packets and another for aggregated packets. Longer bursts leads to longer inactivity in transmission rounds. During this inactivity period, clients can doze off to low power mode.

Wi-Fi device users connect to an AP and contend for the channel; and, if more than one client wake up at the same time, contention gets worse as busy waiting and unnecessary retransmissions occur. Network-assisted power management protocol (NAPman) [11] reduces unnecessary retransmissions by connecting every client to a new copy (virtual) of the AP. Hence, every client gets the feel that the AP is only dedicated to it. Thus, NAPman could avoid collision in spite of having multiple clients connected to an AP. AP virtualization might solve the problem of contention among clients to some extent; however, it is no match for multiple physical APs in the domain. Although the NAPman [11] is providing the clients with virtual APs, there is a limit of the maximum number of virtual APs (4) per NAPman AP (physical), and when NAPman runs out of this number, it starts multiplexing among the clients, increasing media contention significantly.

SleepWell [12] uses the analogy "going to work late and returning home early"; this gives the driver advantage of avoiding the rush hours. It transfers data when the network is least competitive and schedules user's wake-up times accordingly. An AP monitors ongoing traffic of nearby APs and transmits its own traffic only when no one else is using the channel. The SleepWell works well with multiple APs in a wireless vicinity when there are no hidden terminals. However, hidden terminals jeopardize the scheduling adding unfairness to the system and introducing overlapping transmission intervals among the APs. Hence, the time slots of different clients are multiplexed, and they remain awake even if they have no part in the transmission. The SleepWell [12] ensures scheduling of dynamic situations using initially equal workload and gradually giving them weights.

Some latest innovations of Internet such as software-defined networking and network virtualization have been introduced in recent works [25, 26] which state that software-defined networking is more efficient than regular mobile and wireless networking. Green mobile networks, mobile cloud computing, and context aware mobile networking upsurges the capabilities and energy-efficiency of resources [27–30]. Nowadays, cellular networks are moving towards distributed architecture from conventional centralized system with the introduction of femtocells consisting of some self-organizing rules and energy-efficient resource allocation algorithms [31, 32].

The novelty of our proposed PowerNap algorithm stems from the following differences between SleepWell and PowerNap. *Firstly*, we give weighted fair share from the very beginning, i.e., we estimate traffic loads of all APs in a distributed manner and distribute the transmission intervals to the contending APs according to this weighted estimation. *Secondly*, to avoid transmission overlapping, we use undershooting, e.g., the least traffic load estimation; the estimation process involves one hop neighborhood information exploitation only. *Thirdly*, the undershooting again helps PowerNap to reduce transmission overlapping among the APs unlike SleepWell and thus promotes fairness; and *fourthly*, the unused portions (created due to undershooting) of transmission rounds are also distributed in a weighted manner, and thus, it maintains good level of network throughput. *Finally*, we also address the existing hidden terminal problems in our algorithm.

3 Network model and PowerNap design principle

Most of the power saving methodologies consider single AP in a vicinity. However, the incremental growth of Wi-Fi-supported devices and dense deployment of APs have given birth to the problem of interference among the multiple APs which are competing for the same channel and subsequently clients experiencing longer latency.

3.1 Network topology and assumptions

We represent the wireless network as a graph $G(V, E)$, where APs are the vertices and an edge is placed between two APs if they are within each others transmission range.

We consider several APs in a wireless vicinity, as shown in Fig. 1. Each AP sends its ID, time stamp, and workload update messages periodically to its neighbors. In Fig. 1, AP_A has two neighbors, namely, AP_B and AP_E with whom it will share the abovementioned information. However, AP_C is not in the transmission range of AP_A, and thus, AP_A will not share any information with AP_C. Initially, every AP calculates how long it is allowed to transmit in a beacon interval. We call it *weighted fair share*, which is a workload estimation based on the workload demand of a particular AP and its neighbors. Initially, the client or Smartphone sends its workload demand to the AP that estimates its total workload. Using neighborhood discovery, an AP intercepts the workloads of other APs in its range (e.g., one hop neighborhood) and calculates its own share in beacon interval or the transmission ring. We use the terminologies *beacon interval* and *transmission ring* interchangeably throughout this paper. Figure 2a shows the transmission ring produced by AP_A. Figure 2a depicts that AP_C has no contribution in the transmission ring, but it will definitely have impact on AP_A's weighted fair share as it has contribution in AP_B's weighted fair share and AP_B has impact on AP_A. When we look at the transmission ring produced by AP_E in Fig. 2b, we see that AP_A

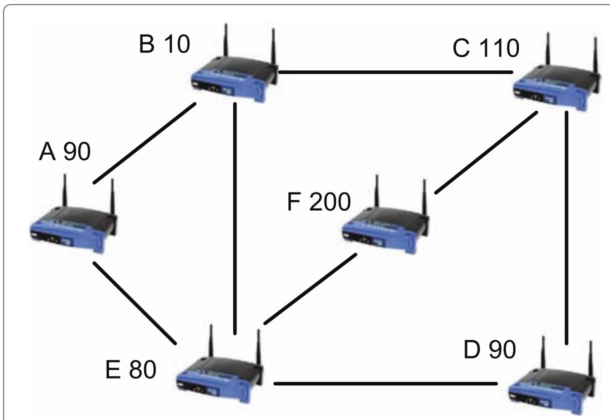


Fig. 1 Sample network environment

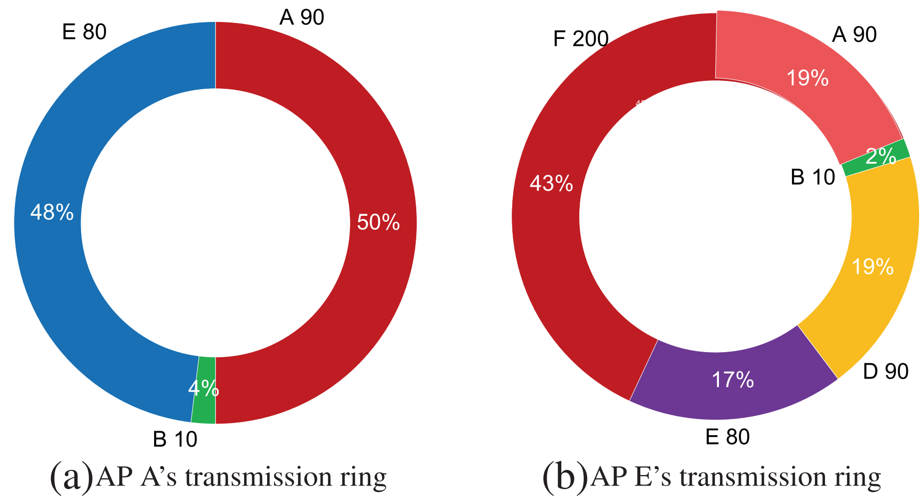


Fig. 2 Transmission rings of APs (**a**) A and (**b**) E without PowerNap implementation

has a fair share of 19 time units whereas in Fig. 2a, it had 50 time units which is clearly wrong, and it will add contention to the network and hamper other APs' transmissions. Our proposed PowerNap scheduling algorithm minimizes such contention by intelligently deciding the optimal transmission time and time-span for a given AP to communicate with its clients. Therefore, the AP and its clients require synchronization.

3.1.1 AP-client clock synchronization

As the client needs to know in advance when it will have to wake up in order to receive packets from the AP, the PowerNap has to implement clock synchronization between a client and its associated AP. The IEEE 802.11 AP broadcasts beacon frames at regular intervals that consist of traffic information (beacon interval, time stamp, and traffic indication map (TIM), estimated payload, etc.) associated to the AP. The client device is logically synchronized with the AP using beacon frame's time stamp attribute. Our PowerNap configures the APs to send their starting and finishing times to the clients in reference to the current time. The clients adjust their wake up time using the time stamp information in the TIM from its AP.

3.2 PowerNap design principle

The design principle behind PowerNap is the transmission of one AP minimally interferes with the transmission of another, and each AP gets its weighted fair share for the media access. The APs will be staggered throughout the transmission round for this purpose. The distributed nature of the algorithm assists the APs to minimize computational latency.

3.2.1 Avoidance of unfairness

All the APs are using the same channel, but the network graph of the APs' connectivity is not complete bipartite. Therefore, an AP, which is unaware of the presence of some other APs, would not be able to calculate its true fair share in a single beacon interval. The estimated share might be greater than the actual fair share as the AP does not consider the APs other than its one hop neighborhood.

- *Neighborhood assistance for fair share* is "to choose the least fair share calculated for a particular AP." The PowerNap allows every AP_j, that receives the workload of another AP_i, to calculate the fair share for AP_i. So the fair share for AP_i in the beacon interval $\mathscr{B}$ is being calculated by all of its one hop neighbors'. Therefore, AP_j, while calculating the fair share for AP_i, would consider its own one hop neighborhood. Hence, it might consider some of the APs that were invisible to AP_i. The neighbor APs send their estimations to AP_i. Now, AP_i has a pool of fair share values. By selecting the least estimated share, it avoids unfairness as much as possible.
- *Undershooting of workload* is used as an extra precaution to avoid unfairness. Though it is unlikely that after the neighborhood assistance for fair share calculation, there would still be situations where APs will spill into the next beacon interval. But if the network is dense and the previous step fails to include every AP in the domain, such a spilling will occur. Thus, using an undershooting parameter to scale down the fair share would be a better idea;

otherwise, the convergence of a steady state is unlikely to happen any time soon.

3.2.2 Channel utilization and hidden terminal

Undershooting the workload might create "unused portion" (no AP is scheduled to transmit on that portion of the ring). This arises the problem of low channel utilization. Hence, we undertook allocation of unused portion. The PowerNap fairly allocates the unused portion of previous round to the contending APs.

Due to hidden terminals, the packets at an AP might get dropped and subsequent retransmissions will occur. The channel could have used this time to transmit data packets; instead, they get stuck in unnecessary traffic generation and propagation. Therefore, it is crucial to address hidden terminal problem in PowerNap. The notations used in this paper are listed in Table 1.

4 PowerNap architecture

In this section, the architectural components and detailed operational procedure of PowerNap have been described. PowerNap minimizes "transmission window overlapping," i.e., multiple APs in the same vicinity start or continue to transmit packets to their clients at the same time. The transmission window overlapping causes prolonged stay in high-power transmission mode for the clients, of which a small amount of time is spent for actual communication. The PowerNap exploits this shortcoming by redesigning it for non-overlapping transmission windows. It aims to stagger the transmission window of APs from one another, so their clients do not suffer for the contentions of their APs. PowerNap has three functional components: *Initialization, Data Transmission Module, and Dynamic Rescheduling*. PowerNap is installed on the APs; clients are completely unaware of these changes and calculations happening on the AP side, and thus, the proposed system is transparent from the clients' perspective.

Table 1 Notations of PowerNap algorithm

U	Set of all clients in the network
U_i	Set of all clients associated with AP i
AP	Access point
ζ_i	Initial workload of AP i
$\mathscr{W}_i$	Weighted fair share of AP i
$\mathbb{W}_i$	Minimal fair share of AP i
B_u	Bit rate of client $u \in U$
$\mathscr{F}_i$	Fair share of AP i in current $\mathscr{B}$
L_k	Workload of AP k
N_j	One hop neighbors of AP_j
$L_{n[i]}$	Workload of neighbor i of AP_k
C_i	Capacity of AP i
n	Number of APs in the network
$\mathscr{W}_{i,j}$	Weighted fair share of AP i computed by AP j

4.1 Initialization

Initially, every AP calculates its initial workload by exploiting neighborhood information and network characteristics, e.g., channel characteristics, AP's own capacity [33], etc. and traffic demands of associated clients.

4.1.1 Initial traffic workload

Workload of an AP is defined as cumulated expected data rate of the associated clients. Clients pass their bit rate (B) demand to their AP for various applications such as Internet telephony, text message service, video streaming, file transfer, bulk data transfer, etc. A "Hello" packet is sent from the client to all the APs in its transmission range, and important parameters such as bit rate demand, signal strength demand, and reliability levels are specified in the "Hello" packet for AP selection [34]. Using these information, AP_i calculates its initial workload, $\zeta_i = \frac{\sum_{\forall u \in U_i}(B_u)}{C_i}$.

4.1.2 Workload information sharing

An AP shares its initial workload among the neighboring APs to build an understanding of the network environment. This workload information sharing is triggered at an interval determined by the dynamic rescheduling component. Dynamic rescheduling is discussed in detail in Section 4.3.

4.1.3 Weighted fair share

The term weighted fair share ($\mathscr{W}$) defines the share of an AP in each transmission round with respect to the workload of the APs in one hop neighborhood. Weighted workload determines the share of each AP in a transmission round, implementing fair media access. All the APs deal with huge amount of traffic, and one AP is only aware of its one hop neighborhood, so the $\mathscr{W}$ is fair locally. But, when we consider the entire network, it will introduce unfairness to APs who are situated at distance more than one hop. So we ask the AP's to calculate weighted fair share for its neighbors too; in this way, we have a broader perspective of the network.

4.1.4 Minimal weighted fair share

Every AP calculates its neighbor's $\mathscr{W}$ along with $\mathscr{W}$ for itself. This minimizes the probability of hidden terminals affecting the fair share calculation adversely since AP's familiarity with network increases the fairness and performance. Next, all the APs send what they have calculated for one particular AP_i, from which AP_i selects the least.

We call this the minimal weighted fair share ($\mathbb{W}$) of AP_i. Table 2 shows the fair share selection model.

4.1.5 Undershooting mechanism

Because of the unpredictable nature of Internet traffic [35] and the fact that Internet traffic is increasing everyday plus a given AP at any given time would not have knowledge of the entire network, one AP is likely to spill into or overlap with other AP's transmission time; even worse, it might deprive others from transmitting. To avoid such situation, undershooting of $\mathbb{W}$ is used. Undershooting is the process of employing less than the estimated value. A carefully chosen threshold α will serve this purpose. By scaling down the transmission time, we can have a fair transmission environment at the cost of one setup transmission round.

PowerNap scales down the values extracted from Table 2 using α, the undershooting parameter. Let us assume $\alpha = 75\,\%$; the minimal fair share after undershooting for the network of Fig. 1 is shown in Table 3. The Fig. 3a shows the transmission ring after undershooting.

4.2 Data transmission

Now, the APs transmit the data packets they have queued over time. The media accessing order of APs is controlled by their *back-off timers*, which is inversely proportional to their traffic weights. Every AP transmits a "free-to-transmit" message after the completion of its transmission to announce the availability of the channel to other APs.

After the *back-off timer* expires, an AP scans for a "free-to-transmit" message. On the reception of this message, the AP starts to transmit the data packets to its clients. At the end of every transmission round, the channel has the "free-to-transmit" message of an AP, which is transmitted last in this round. This message will be heard by the first AP of the next transmission round. All the APs embed End-Time (transmission completion) in "free-to-transmit" message. The End-Time received by the first AP is used as the finishing time of the previous beacon interval. End-Time is also used in unused portion allocation.

Table 2 Weighted fair share calculation

AP	A	B	C	D	E	F
A	50	6	–	–	44	–
B	31	3	38	–	28	–
C	–	27	22	–	49	–
D	–	–	39	32	29	–
E	19	2	–	19	17	43
F	–	–	28	–	21	51
$\mathbb{W}$	19	2	22	19	17	43

Table 3 Fair share calculation

AP	$\mathscr{F}$ after undershooting	$\mathscr{W}$ after unused portion allocation
A	4.75	15.57
B	0.5	1.64
C	5.5	18.04
D	4.75	15.57
E	4.25	13.93
F	10.75	35.25

4.2.1 Unused portion allocation

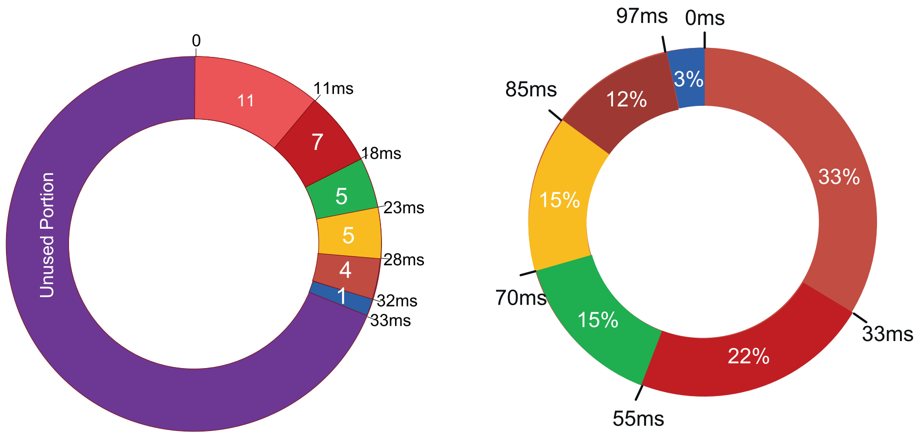
The employment of undershooting might cause the channel to be in idle state for certain duration, for example when all the APs finish transmitting and the beacon interval is yet to finish. We call this idle period "unused portion" of the $\mathscr{B}$. To utilize the channel efficiently, we develop an unused portion allocation scheme. The idea is to consider the mean time between the End-Time and the starting of the current beacon interval. Table 3 shows the $\mathscr{F}$ of the APs after unused portion allocation.

All APs need to synchronize by using beacon packets, and workload sharing or weighted fair sharing are all dependent on message passing among the APs. So, loss of packets will put strain on PowerNap. Such fault tolerant issues related to beacon packet loss and performance optimality are kept as future works.

4.2.2 Tradeoff between "undershooting" and "unused portion allocation"

Apparently, it seems undershooting and unused portion allocation mechanisms neutralizes each other and have no significant impact on the scheduling. But in fact, implementation of undershooting within a beacon interval creates scope for fairness, as the mere allocation of weighted fair share might cause spill in the next beacon interval if the $\mathscr{B} < \sum_{\forall AP} \mathscr{W}$. To accommodate all the APs in the current $\mathscr{B}$, we employ undershooting. The undershooting is dependent on the metric α. This will generate local optimal shares for each AP; hence, it might generate some unused portion in the $\mathscr{B}$. Such unused portion in $\mathscr{B}$ leads to low channel utilization and needs to be assigned to the participating APs in a weighted manner, and thus, the unused portion allocation scheme is required. Details of undershooting parameter α is discussed in Section 4.4.

The unused portion is divided among the APs according to $\mathbb{W}$ weights of APs ensuring fairness. Figure 3b shows the transmission ring after unused portion allocation. Algorithm 1 shows the operation principles of PowerNap algorithm.



(a)Undershooting mechanism (b)Unused portion allocation

Fig. 3 a Undershooting mechanism and **b** Unused portion allocation

Algorithm 1 PowerNap Scheduling Algorithm *for each AP*

1. **Initialization:**
2. Initial workload of AP i, $\zeta_i = \frac{\sum_{\forall u \in U_i} B_u}{C_i}$
3. **Workload minimization :**
4. Weighted Fair Share of AP i, $\mathscr{W}_i = \frac{\zeta_i}{\zeta_i + \sum_{\forall j \in N_i} \zeta_j}$
5. Weighted Fair Share of AP i computed by AP j, $\mathscr{W}_{i,j} = \frac{\zeta_i}{\zeta_j + \sum_{\forall i \in N_j} \zeta_i}$
6. Minimal Fair Share of AP i, $\mathbb{W}_i = \min\{\mathscr{W}_i, \mathscr{W}_{i,j}\}$
7. Fair Share of AP i in current $\mathscr{B}$, $\mathscr{F}_i = \mathbb{W}_i \times \frac{100-\alpha}{100}$
8. $timer = \frac{1}{\mathscr{F}_i}$
9. **Transmission:**
10. **if** ($timer$ expires AND AP i receives $free-to-transmit$ message) **then**
11. AP i starts communication
12. reset end_time to current time
13. **end if**
14. **Unused portion allocation:**
15. **if** new beacon interval starts **then**
16. $\mathscr{F}_i = \frac{\mathscr{F}_i \times \mathscr{B}}{end\_time}$
17. **end if**

Every AP records its wake up time and transmission duration and advertises these to its clients and use these values for further communications. Thus, whenever an AP needs to communicate with its clients, it will be awake and ready to respond, except for occurrence of dynamic traffic changes. Dynamic conditions are handled through rescheduling in PowerNap.

4.3 Dynamic rescheduling

Traffic arrival and departure in a typical Wi-Fi network follows exponential distribution. Three major incidents controls dynamic rescheduling [36]: arrival of new AP, departure of an existing AP, and substantial change in traffic loads. Rescheduling can be periodic or triggered by interrupt. All the APs continuously scans the Wi-Fi spectrum for any demand changes, and if the upper limit for rescheduling is met, we move to the initialization phase again and repeat the undershooting process. Inclusions or failures of APs are interrupt-triggered rescheduling situations. We set β as rescheduling threshold, an upper limit by which it is determined that after β % change in user data rate, PowerNap will be rescheduled. Analysis of rescheduling parameter β will be addressed in future.

4.4 Analysis of undershooting parameter, α

The weighted fair share of an AP greatly depends on its number of neighbor APs and their traffic workloads. Let we represent the Wi-Fi network as a connected graph, $\mathbb{G}(\mathbb{V}, \mathbb{E})$, where, $\mathbb{V}$ is the set of vertices of all APs and $\mathbb{E}$ is the set of all edges. For a complete bipartite graph, i.e., the graph forms a clique, the number of edges, $E = |\mathbb{E}|$ is calculated as,

$$E = \frac{V(V-1)}{2}$$
$$\frac{V^2 - 2E}{V} = 1, \qquad (1)$$

where, $V = |\mathbb{V}|$ is the total number of vertices in the graph. Notice that, for a clique network, all APs have the traffic workload information of all others, which is an ideal condition for our proposed PowerNap protocol. In this case, the weighted fair share of all the APs will fill up the total transmission ring and thus the value of undershooting parameter α will be equal to 1.

However, in practical networks, graphs will often be non-clique and the Eq. 1 will not be held; it changes to

$$\frac{V^2 - 2E}{V} > 1. \tag{2}$$

In this case, the value of α for an AP will be greater than 1 and its value will be determined by scaling factors of an AP and its neighbors. The scaling factor of an access point i will be determined as follows,

$$\psi(i) = \frac{\zeta_i}{\sum_{\forall j \in N_i} \zeta_j + \zeta_i} + \sum_{\forall j \in N_i} \frac{\zeta_j}{\sum_{\forall k \in N_j} \zeta_j + \zeta_j} \tag{3}$$

Each AP broadcasts its $\psi(i)$ into the network and all the APs compute α as follows,

$$\alpha = \left\{ 1 - \frac{\mathscr{B}}{\max\{\psi(1), \psi(2), \ldots, \psi(N_n)\}} \right\} \times 100. \tag{4}$$

5 Performance evaluation

In this section, we study the performances of the proposed PowerNap algorithm, IEEE 802.11 PSM [13] and SleepWell [12] in terms of energy consumption, network throughput, fairness, and protocol operation overhead, carried out in ns-3 [17].

5.1 Simulation environment

We set up a dense Wi-Fi network, where all APs can hear their one hop neighbors' transmission, and the number of clients under each AP is randomly varied with dynamic traffic loads to test the robustness of our algorithm. APs use 1 Mbps as basic bit rate, in which control packets are sent and data transmission is started. We simulate 3 GB of bulk data for simulation data collection purposes. The default beacon interval is 100 ms and the default AP number is 6. The clients and the APs use *StaWifiMac* and *ApWifiMac* as their MAC layer protocols and *RandomDirection2dMobility* and *ConstantPositionMobility* as their mobility models, respectively. The *RandomDirection2dMobility* model uses random values for speed, pause time, direction, and acceleration of the mobile nodes, which is corresponding to the real-life Wi-Fi network scenario. The channel properties such as delay loss model, propagation delay model, data rate, and channel characteristics are defined using *YansWifiPhy* model. The simulation is run for 300 s. For each graph data points, we take average value taken from 10 different simulation runs, with different seed values. In Table 4, we have listed the important parameters and their values/models used in the simulation.

Table 4 Simulation parameters

Name of the parameter	Value or model used
Access point MAC layer model	ApWifiMac(IEEE 802.11b)
Client MAC layer model	StaWifiMac(IEEE 802.11b)
Access point mobility	ConstantMobility-Model
Client MAC layer model	RandomDirection2-dModel
Area	1500×1500 m^2
Number of AP	6
Number of client	$2 \sim 50$
Physical layer model	YansWifiPhy model
Transmission range of APs	50 m
Beacon interval	100 ms
Data rate	802.11b maximum raw data rate of 11 Mbps
Channel bandwidth	2.4 GHz
Simulation time	300 s

5.2 Clients energy modes

For energy-saving, our proposed PowerNap algorithm takes advantage of various power levels of today's Wi-Fi enabled devices. Most devices implement a few power levels for seamless networking operation. Usually, there are five different power levels [37], as shown in Table 5.

By tailoring the time spent in each mode, PowerNap cuts down on the energy expenditure. Transmission mode facilitates the APs to transmit their data packets. Amount of time spent here is decided by the client's bit rates and its workload. Sleep mode consumes the least amount of energy as the entire Wi-Fi radio spectrum runs on low power. Internal processing such as timer monitoring and routine tasks are only performed in this mode. The overhear mode is considered a potential source of energy wastage because even after finishing transmission, it keeps the entire circuitry up and listens to every packet the AP is transmitting. In listening mode, a client only listens for beacon frames; it wakes up periodically and receives beacon traffic. PowerNap is configured to choose listening mode over overhear mode. Hence, it saves valuable battery power of the clients. In PSM, it shifts to overhear mode right after transmission mode and stays there for a long duration expecting a high power transmission again shortly. However, in PowerNap, a client will only be awaken when its AP is transmitting. Moreover, one client

Table 5 Various power modes of Samsung Galaxy S2

Mode	Energy (in mW)
Deep sleep	10
Light sleep	120
Wake up from deep sleep	250
High power transmission	600
Overhear	400

will receive one transmission chance in one beacon interval. This makes it easier for a client to go to listening mode right after the high power transmission is finished.

5.3 Performance metrics

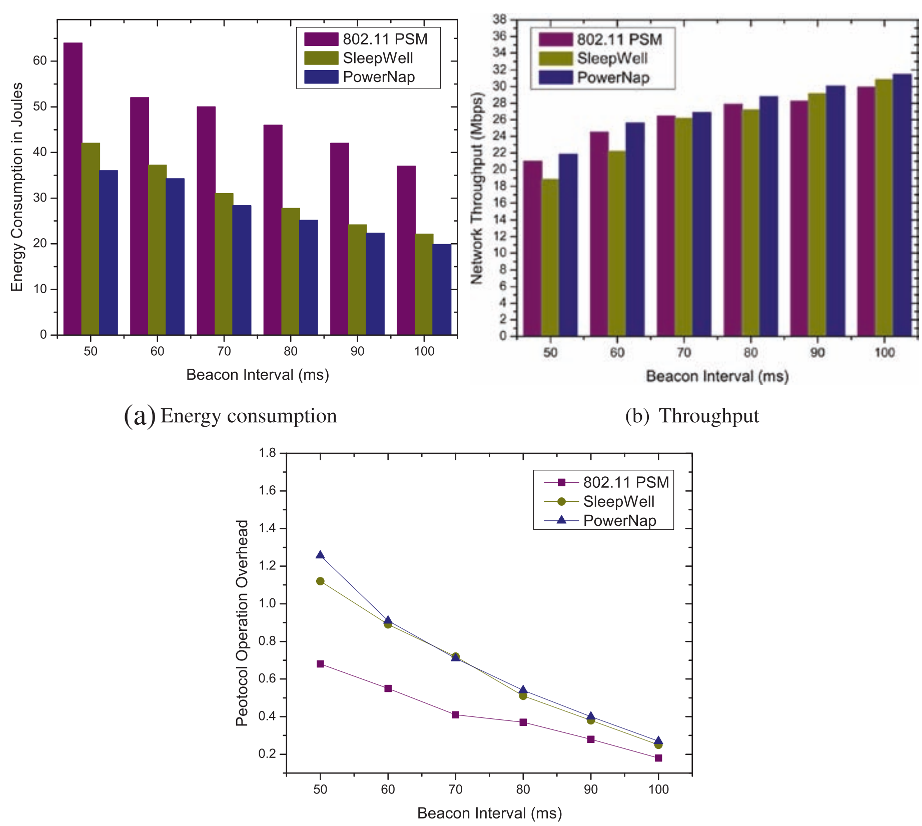
- *Energy consumption* is measured as the average amount of energy consumed by each client during the simulation period. We have considered energy consumptions due to data transmissions, receptions, overhearing, idle, and sleep periods.
- *Network throughput* measurement and comparison is of foremost importance as most of the energy-conservation mechanisms tend to compromise network throughput to conserve more energy. It is measured as the average amount of data bytes transferred by each AP per unit time. The higher the value is, the better the protocol performance is.
- *Protocol operation overhead* is measured as the ratio of the average number of control packets transmitted by the APs during the simulation period and network throughput. The control packets include hello packets, free-to-transmit message, clock synchronization packets, etc.
- *Jain's fairness index* [38] has been used for measuring the fairness level of the studied protocols based on AP.

5.4 Simulation results

5.4.1 Impacts of beacon intervals

The beacon interval has great impacts on the performances of the studied protocols, as shown in Fig. 4. Since our proposed PowerNap algorithm does not have any iterative initialization process like PSM or SleepWell, so the transmission overhead in PowreNap is substantially low. It is also notable that short beacon interval incurs immense state switching energy. Our proposed PowerNap wastes energy for state switching only while both PSM and SleepWell have to account for the energy needed for rescheduling too. Figure 4a shows that the average power consumption per client in PowerNap is much less compared to the SleepWell and IEEE 802.11 PSM.

Figure 4b shows that the achieved throughput of PowerNap is better than IEEE 802.11 PSM and SleepWell. In both PowerNap and IEEE 802.11 PSM, the beacon interval has a significant contribution in network throughput as the increased value of beacon interval produces less overhead and the network throughput increases. As stated



(a) Energy consumption (b) Throughput (c) Protocol operation overhead

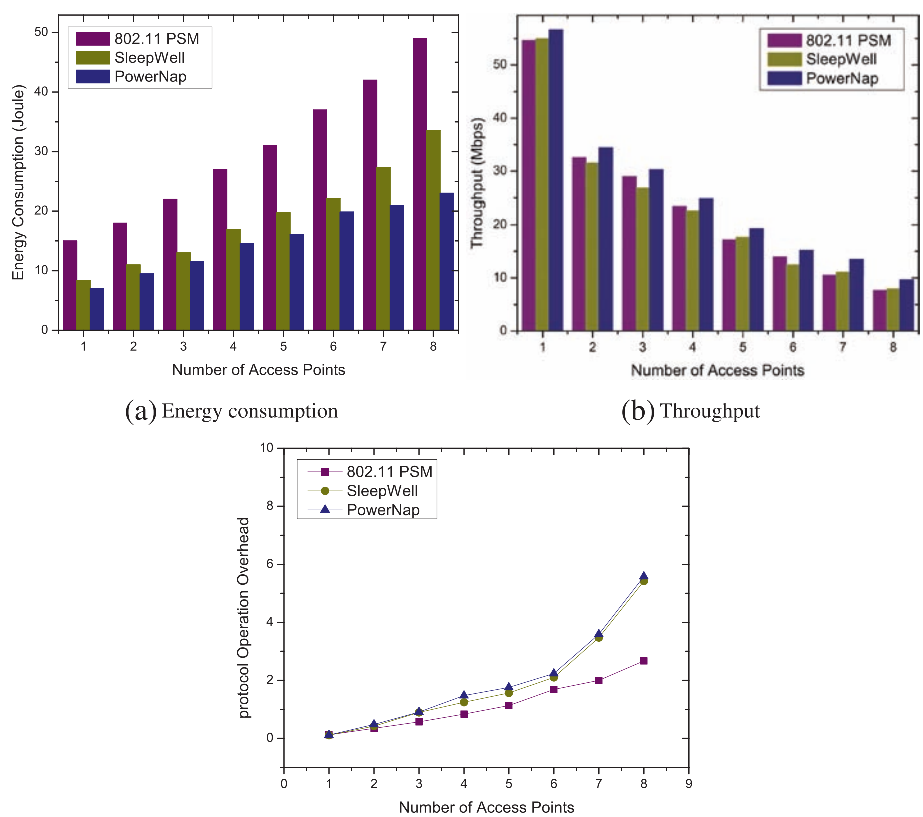
Fig. 4 Performance analysis of varying beacon intervals (**a**) Energy consumption, (**b**) Throughput and (**c**) Protocol operation overhead in our algorithm, it increases the listen time and shortens the overhear time of clients. During listen time, an AP checks if the timer is matured, and during overhear time, an AP overhears the channel for packets. In standard PSM, after waking up, every client overhears the channel for the entire beacon interval; however, in PowerNap, one client overhears the channel only when the AP associated with it is active for transmission. Similarly, the SleepWell incurs much overhead for repeated migration operations, and thus, it keeps the users in overhear mode for longer duration. As expected theoretically, for increasing beacon intervals, the protocol operation overheads of the studied protocols are drecreased, as shown in Fig. 4c. Our proposed PowerNap protocol needs exchange of some additional control packets for its opration, and thus, it incurs more overhead than 802.11 PSM. However, the overall energy consumption is still much less than the studied protocols.

5.4.2 Impacts of number of access points

The energy consumption of all the studied protocols is increased with the number of APs, as shown in Fig. 5a. We observe that our proposed PowerNap algorithm saves substantial amount of energy compared to IEEE 802.11 PSM and SleepWell when the network is dense. The increasing number of APs degrade the performance of 802.11 PSM as it is not designed to deal with multiple APs; both the PSM and SleepWell suffer from unnecessary retransmissions because of hidden terminals, whereas, the scheduling in PowerNap considers all the APs affecting its transmission, and thus, it shows better performance. Figure 5b shows that the achieved throughput of the APs, in all the studied protocols, gradually decreases with the higher number of APs and this is caused due to increased media contention. The PowerNap can achieve better throughput than SleepWell and IEEE 802.11 PSM. The protocol operation overhead increases with the number of APs, as shown in Fig. 5c.

5.4.3 Impacts of transmission rounds

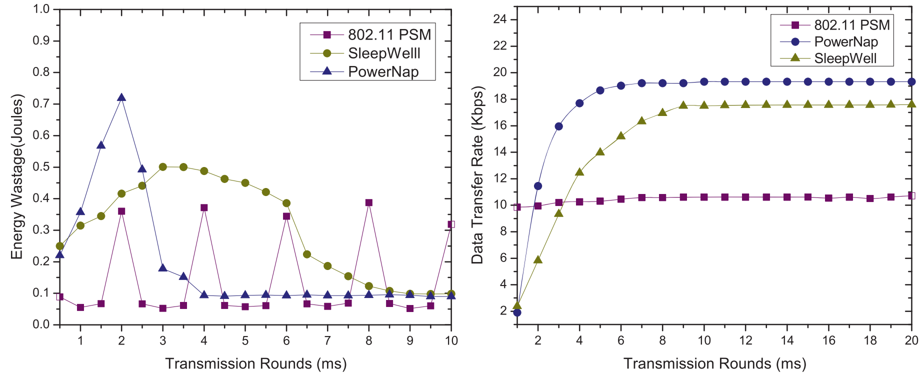
We have also measured the average amount of energy wastage per client due to scheduling for increasing transmission rounds (each of 100 ms), as shown in Fig. 6a. At the start of simulation, PowerNap APs exchange control packets to advertise their payloads, neighbors payload, allocation of unused portion, etc., and thus, its energy



(a) Energy consumption (b) Throughput

(c) Protocol operation overhead

Fig. 5 Performance analysis with varying number of APs (**a**) Energy consumption (**b**) Throughput and (**c**) Protocol operation overhead



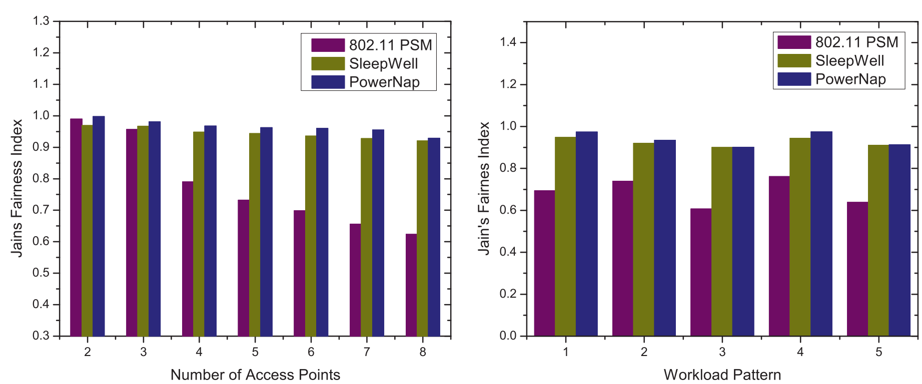
(a) Energy wastage for scheduling (b) Data transfer rate

Fig. 6 a Energy wastage and (**b**) data transfer rate profiles for ongoing transmission rounds

wastage is high. No dynamic rescheduling is performed here, so the energy consumption of PowerNap due to control packets reaches a steady state after the allocation of unused portion. But for PSM, there are spikes in energy levels since at every 300 ms, the state switching control packets introduce additional energy consumption. In Fig. 6b, we have measured the data transfer rate per transmission round, found in the studied protocols. The IEEE 802.11 PSM offers almost steady and linear data transfer rate as it does not have any scheduling overheads. The scheduling process tends to override data packet transfer, so in the first few rounds of PowerNap and SleepWell, data transfer rate is very low as they spend more time in scheduling than transferring data packets. The undershooting mechanism and minimal fair share of PowerNap consume most part of the transmission time in the first few rounds. For SleepWell, the convergence time is lengthier than PowerNap; SleepWell APs incur extensive message passing to their neighbors. We observe that, after a few rounds, the data transfer rates of PowerNap and SleepWell reach at steady state. However, for PSM, the data rate would not increase due to low channel utilization problem in presence of multiple APs. Both PowerNap and SleepWell are very much tolerant of multiple APs unlike 802.11 PSM, and they can adapt to the events where new APs are arrived.

5.4.4 Fairness

We have also measured the Jain's fairness index of the studied protocols for varying number of *APs* and workloads, as depicted in Fig. 7. In Fig. 7a, we observe that the PSM becomes unfair when there are multiple APs in the domain. However, as expected theoretically, PowerNap maintains good level of fairness for all the APs for fair transmissions, even better than SleepWell. In Fig. 7b, we observe the achieved fairness levels of the studied protocols for various workload patterns (WPs), as listed in Table 6. Again, our proposed PowerNap algorithm is proven to be robust enough to handle dynamic traffic fluctuations.



(a) Impacts of number of APs (b) Impacts of workload

Fig. 7 Fairness comparison using Jain–s fairness index, (**a**) impacts of number of APs and (**b**) impacts of workloads

Table 6 Workload pattern (WP) for simulation runs

AP	WP 1	WP 2	WP 3	WP 4	WP 5
AP_A	90	120	20	70	90
AP_B	10	50	300	180	10
AP_C	110	30	250	30	70
AP_D	90	90	10	50	130
AP_E	80	150	20	20	200
AP_F	200	80	40	120	30

6 Conclusions

Rapid drainage of battery power of the smart devices due to extensive Wi-Fi usage with concurrent resource intensive applications is a raising concern among the researchers for quite a sometime now. In this paper, we have designed an access point scheduling algorithm, PowerNap, in order to prevent energy losses due to competition for channel acquirement among the APs. The PowerNap is decentralized and traffic load adaptive. The proposed algorithm is incrementally deployable because it can co-exist with legacy access points, and thus, the current access points can use PowerNap with a view to salvage client's energy in dense Wi-Fi domain with variable number of APs. The simulation results, carried out in ns-3, bear witness that the proposed PowerNap outperforms the state-of-the-art approaches in terms of power consumption and fairness, while maintaining almost the same throughput.

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
The authors would like to extend their sincere appreciation to the Deanship of Scientific Research at King Saud University for its funding of this research through the Research Group Project no. RGP-VPP-281.

Author details
[1]Green Networking Research (GNR) Group, Department of Computer Science and Engineering, University of Dhaka, Dhaka, 1000, Bangladesh. [2]Department of Computer Engineering, Kyung Hee University, Suwon, South Korea. [3]Information Systems Department, College of Computer and Information Sciences, King Saud University, Riyadh, Kingdom of Saudi Arabia.

Received: 8 November 2015 Accepted: 11 January 2016

References

1. H Falaki, R Mahajan, S Kandula, D Lymberopoulos, R Govindan, D Estrin, in *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*. Diversity in smartphone usage (ACM, New York, NY, USA, 2010), pp. 179–194
2. M Anand, EB Nightingale, J Flinn, Self-tuning wireless network power management. Wirel. Netw. **11**(4), 451–469 (2005)
3. GP Perrucci, FH Fitzek, J Widmer, in *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*. Survey on energy consumption entities on the smartphone platform (IEEE, Yokohama, Japan, 2011), pp. 1–6
4. N Balasubramanian, A Balasubramanian, A Venkataramani, in *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement Conference*. Energy Consumption in Mobile Phones: A Measurement Study and Implications for Network Applications (ACM, New York, NY, USA, 2009), pp. 280–293
5. EJ Vergara, S Nadjm-Tehrani, in *Energy Efficiency in Large Scale Distributed Systems*. Energybox: A Trace-Driven Tool for Data Transmission Energy Consumption Studies (ACM, New York, NY, USA, 2013), pp. 19–34
6. MA Hoque, M Siekkinen, JK Nurminen, Energy efficient multimedia streaming to mobile devices⊠ a survey. IEEE Commun. Surv. Tutorials. **16**(1), 579–597 (2014)
7. A Balasubramanian, R Mahajan, A Venkataramani, in *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*. Augmenting Mobile 3g Using Wifi (ACM, New York, NY, USA, 2010), pp. 209–222
8. R Kravets, P Krishnan, in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*. Power Management Techniques for Mobile Communication (ACM, New York, NY, USA, 1998), pp. 157–168
9. Y Zhang, Q Li, in *INFOCOM, 2013 Proceedings IEEE*. Howies: A Holistic Approach to Zigbee Assisted Wifi Energy Savings in Mobile Devices (IEEE, Turin, Italy, 2013), pp. 1366–1374
10. G Anastasi, M Conti, E Gregori, A Passarella, in *Proceedings of the Workshop on Linguistic Theory and Grammar Implementation, ESSLLI-2000*. Saving Energy in Wi-Fi Hotspots Through 802.11 PSM: An Analytical Model, (PARC Palo Alto, USA, 2004), pp. 24–26
11. E Rozner, V Navda, R Ramjee, S Rayanchu, in *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*. Napman: Network-Assisted Power Management for Wifi Devices (ACM, New York, NY, USA, 2010), pp. 91–106
12. J Manweiler, R Roy Choudhury, in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*. Avoiding the Rush Hours: Wifi Energy Management via Traffic Isolation (ACM, New York, NY, USA, 2011), pp. 253–266
13. IW Group, IEEE standard for information technology⊠ telecommunications and information exchange between systems⊠ local and metropolitan area networks⊠ specific requirements⊠ part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 6: wireless access in vehicular environments. IEEE Std. **802**, 11 (2010). https://pdos.csail.mit.edu/archive/decouto/papers/802.11a.pdf
14. K-Y Jang, S Hao, A Sheth, R Govindan, in *Proceedings of the Seventh Conference on Emerging Networking Experiments and Technologies*. Snooze: Energy Management in 802.11n WLANs (ACM, New York, NY, USA, 2011), p. 12
15. Y Agarwal, R Chandra, A Wolman, P Bahl, K Chin, R Gupta, in *Proceedings of the 5th International Conference on Mobile Systems, Applications and Services*. Wireless Wakeups Revisited: Energy Management for VoIP over Wi-Fi smartphones (ACM, New York, NY, USA, 2007), pp. 179–191
16. A Enayet, N Mehajabin, MA Razzaque, CS Hong, in *Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication*. A Power-Aware Distributed Wi-Fi Access Point Scheduling Algorithm (ACM, New York, NY, USA, 2013), p. 41
17. H Wu, S Nabar, R Poovendran, in *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*. An Energy Framework for the Network Simulator 3 (ns-3) (ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, 2011), pp. 222–230
18. ICSLMS Committee, Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. IEEE Std (1997)
19. MA Viredaz, LS Brakmo, WR Hamburgen, Energy management on handheld devices. Queue. **1**(7), 44 (2003)
20. J Sorber, N Banerjee, MD Corner, S Rollins, in *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*. Turducken: Hierarchical Power Management for Mobile Devices (ACM, New York, NY, USA, 2005), pp. 261–274
21. E Shih, P Bahl, MJ Sinclair, in *Proceedings of the 8th Annual International Conference on Mobile Computing and Networking*. Wake on Wireless: An Event Driven Energy Saving Strategy for Battery Operated Devices (ACM, New York, NY, USA, 2002), pp. 160–171
22. A Acquaviva, E Lattanzi, A Bogliolo, in *Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. Design and Simulation of Power-Aware Scheduling

Strategies of Streaming Data in Wireless LANs (ACM, New York, NY, USA, 2004), pp. 39–46
23. K-H Jung, J-P Jeong, Y-J Suh, Sleeping mobile ap: a novel energy efficient wifi tethering scheme. Wirel. Netw. **21**(3), 963–980 (2015)
24. R Palit, K Naik, A Singh, in *2011 7th International Wireless Communications and Mobile Computing Conference (IWCMC)*. Impact of Packet Aggregation on Energy Consumption in Smartphones (IEEE, Istanbul, Turkey, 2011), pp. 589–594
25. M Yang, Y Li, D Jin, L Zeng, X Wu, AV Vasilakos, Software-defined and virtualized future mobile and wireless networks: a survey. Mob. Netw. Appl. **20**(1), 4–18 (2014)
26. X Jia, Y Sikya, SBMP: An SDN-based mobility management protocol to support 828 seamless handover, MS thesis, Department of Electrical Engineering and Information Systems, University of Tokyo (August, 2015)
27. X Wang, AV Vasilakos, M Chen, Y Liu, TT Kwon, A survey of green mobile networks: opportunities and challenges. Mob. Netw. Appl. **17**(1), 4–20 (2012)
28. MR Rahimi, J Ren, CH Liu, AV Vasilakos, N Venkatasubramanian, Mobile cloud computing: a survey, state of art and future directions. Mob. Netw. Appl. **19**(2), 133–143 (2014)
29. MR Rahimi, N Venkatasubramanian, AV Vasilakos, in *Cloud Computing (CLOUD), 2013 IEEE Sixth International Conference On*. Music: Mobility-Aware Optimal Service Allocation in Mobile Cloud Computing (IEEE, Santa Clara, CA, USA, 2013), pp. 75–82
30. Z Feng, Y Zhu, Q Zhang, LM Ni, AV Vasilakos, in *INFOCOM, 2014 Proceedings IEEE*. Trac: Truthful Auction for Location-Aware Collaborative Sensing in Mobile Crowdsourcing (IEEE, Piscataway, NJ, USA, 2014), pp. 1231–1239
31. D López-Pérez, X Chu, AV Vasilakos, H Claussen, Power minimization based resource allocation for interference mitigation in OFDMA femtocell networks. IEEE J. Sel. Areas Commun. **32**(2), 333–344 (2014)
32. D López-Pérez, X Chu, AV Vasilakos, H Claussen, On distributed and coordinated resource allocation for interference mitigation in self-organizing lte networks. IEEE/ACM Trans. Networking (TON). **21**(4), 1145–1158 (2013)
33. S Vasudevan, K Papagiannaki, C Diot, J Kurose, D Towsley, in *Proceedings of the 5th ACM SIGCOMM Conference on Internet Measurement*. Facilitating Access Point Selection in IEEE 802.11 Wireless Networks (USENIX Association, Berkeley, CA, USA, 2005), pp. 26–26
34. F Xu, X Zhu, CC Tan, Q Li, G Yan, J Wu, Smartassoc: Decentralized access point selection algorithm to improve throughput. IEEE Trans. Parallel Distrib. Syst. **24**(12), 2482–2491 (2013)
35. H Falaki, D Lymberopoulos, R Mahajan, S Kandula, D Estrin, in *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*. A First Look at Traffic on Smartphones (ACM, New York, NY, USA, 2010), pp. 281–287
36. T Armstrong, O Trescases, C Amza, E de Lara, in *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services*. Efficient and Transparent Dynamic Content Updates for Mobile Clients (ACM, New York, NY, USA, 2006), pp. 56–68
37. A Garcia-Saavedra, P Serrano, A Banchs, G Bianchi, in *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies*. Energy Consumption Anatomy of 802.11 Devices and its Implication on Modeling and Design (ACM, New York, NY, USA, 2012), pp. 169–180
38. R Jain, A Durresi, G Babic, Throughput fairness index: an explanation. Technical report, Tech. rep., Department of CIS, The Ohio State University (1999)