

# Life-Add: Lifetime Adjustable Design for WiFi Networks with Heterogeneous Energy Supplies

Shengbo Chen<sup>\*§</sup>, Tarun Bansal<sup>†§</sup>, Yin Sun<sup>\*§</sup>, Prasun Sinha<sup>†</sup> and Ness B. Shroff<sup>‡</sup>

<sup>\*‡</sup>Department of ECE, The Ohio State University

<sup>†‡</sup>Department of CSE, The Ohio State University

Email: {chens,shroff}@ece.osu.edu, {bansal,prasun}@cse.ohio-state.edu, sunyin02@gmail.com

<sup>§</sup>Co-primary authors

**Abstract**—WiFi usage significantly reduces the battery lifetime of handheld devices such as smartphones and tablets, due to its high energy consumption. In this paper, we propose “Life-Add”: a Lifetime Adjustable design for WiFi networks, where the devices are powered by battery, electric power, and/or renewable energy. In Life-Add, a device turns off its radio to save energy when the channel is sensed to be busy, and sleeps for a random time period before sensing the channel again. Life-Add carefully controls the devices’ average sleep periods to improve their throughput while satisfying their operation time requirement. It is proven that Life-Add achieves near-optimal proportional-fair utility performance for single access point (AP) scenarios. Moreover, Life-Add alleviates the near-far effect and hidden terminal problem in general multiple AP scenarios. Our ns-3 simulations show that Life-Add *simultaneously* improves the lifetime, throughput, and fairness performance of WiFi networks, and coexists harmoniously with IEEE 802.11.

## I. INTRODUCTION

WiFi is one of the most successful wireless Internet access technologies, which has become prevalent for handheld mobile devices, such as smartphones and tablets. One major concern of the current IEEE 802.11 WiFi design is its high energy consumption. WiFi, along with GPS and Bluetooth, uses a significant amount of energy, which causes rapid drops in battery level, especially for smartphones. In fact, it has been reported that the IEEE 802.11 WiFi design may increase the smartphone’s energy consumption by up to 14 times [1].

Currently, there are two categories of works that focus on prolonging the operation time of WiFi devices. The first class aims at reducing the energy consumption of WiFi radios, e.g., [1]–[3]. The second class uses a renewable solar charger or a portable battery to provide mobile energy replenishment without being confined to the immovable electric power sources on the wall, e.g., [4], [5]. Beyond these, it would be of great interest to design lifetime-tunable WiFi devices where the throughput performance can be also improved.

In this paper, we propose *Life-Add*: a Lifetime Adjustable design for WiFi networks, where the devices are powered by battery, electric power, and/or renewable solar energy. In this design, a device senses the channel availability when it has a packet to send. If the channel is sensed to be busy, it goes to sleep for an exponentially distributed random period of time and then wakes up to sense the channel availability again. If the channel is idle, it transmits the packet. Life-Add carefully controls the average sleep periods of the WiFi devices

to improve their throughput and fairness performance, while satisfying their operation time requirement. We first formulate a proportional-fair network utility maximization problem for asynchronous WiFi networks in single AP scenarios. Unlike traditional synchronized wireless networks, such as cellular networks, the channel access probabilities of the WiFi devices are coupled due to channel contentions. As a result, the attained network utility maximization problem turns out to be a difficult non-convex optimization problem. However, we manage to develop a low-complexity solution and prove that it is near-optimal to the problem. In general multiple AP scenarios, due to the complicated interference environment, the WiFi networks suffer from the well-known “*near-far effect*” and the “*hidden terminal problem*”. Life-Add introduces device collaboration to alleviate the near-far effect and leverages congestion control to handle the hidden terminal problem.

The contributions of this paper are summarized as follows:

- We propose a new model for the WiFi transmission behaviors, and characterize the throughput and energy consumption performance of each device. Thanks to the memoryless property of the exponential sleep period distribution, the attained performance metric functions are quite simple. This forms the basis for the development of an optimization framework, which leads to our control strategy.
- We formulate a non-convex proportional-fair utility maximization problem for the single AP scenarios, and develop a simple solution, i.e., Life-Add, to control the average sleep time periods. We prove that Life-Add is near-optimal, if the carrier sensing time is negligible compared to the packet-plus-ACK transmission time. This condition is satisfied for practical WiFi packet transmission parameters; see the Remark in Section V-D for details.
- We generalize Life-Add to deal with the near-far effect and the hidden terminal issue heuristically. Our key contribution here is to develop a new device collaboration scheme, which improves the throughput and fairness performance. In addition, the Life-Add design is simple and easy to implement.
- Our ns-3 simulations show that Life-Add simultaneously improves the lifetime, throughput, and fairness performance of WiFi networks. A brief performance comparison

**TABLE I.** Performance comparison of different WiFi designs.

	IEEE 802.11b	IEEE 802.11b with RTS-CTS	Life-Add
Avg Lifetime	147.87 min	168.79 min	252.45 min
Avg throughput	0.9568 Mbps	1.0338 Mbps	1.41 Mbps
Jain's fairness index [6]	0.427	0.402	0.691
successful ACK	68.3%	98.6%	87.45%

between Life-Add and IEEE 802.11b is provided in Table I for the simulation scenario of Fig. 6. We also show that Life-Add can coexist harmoniously with IEEE 802.11: if some devices upgrade from IEEE 802.11 to Life-Add, the throughput performance improves for both the devices switching to Life-Add and those sticking to IEEE 802.11.

Compared to IEEE 802.11, first, the lifetime improvement of Life-Add is due to turning off the radio during the backoff period. Second, by controlling the transmission parameters carefully and introducing collaboration among the devices, Life-Add has a smaller packet collision probability and thus leads to better throughput performance. Finally, by providing higher priorities to the low-throughput devices, Life-Add can improve the fairness performance.

## II. RELATED WORKS

Performance analysis of carrier sensing multiple access (CSMA) schemes for WiFi networks have been widely investigated since its seminal paper [7]. However, the two dimensional Markovian chain model in [7] makes it difficult to control the WiFi transmissions. Recently, a queue based scheduling scheme called Q-CSMA, was proven to be throughput optimal [8]. These studies focus on throughput enhancement, assuming the WiFi radio is turned on all the time for sensing, transmitting, or receiving, which is not energy efficient.

Energy efficient scheduling schemes have been developed for synchronized wireless networks (see the recent papers [9]–[12] and the references therein). These works focused on slotted wireless networks, such as cellular networks. Therefore, they are not applicable to asynchronous WiFi networks.

Sleep-wake scheduling schemes have been developed for wireless sensor networks to prolong the battery lifetime (see [13]–[15] and the references therein). While these schemes apply to asynchronous wireless networks, their focus is on light traffic sensor networks where packet collisions seldom occur. However, packet collision is one of the most serious problems in WiFi networks.

Recently, energy efficient WiFi designs have received significant research attention due to the emerging handheld WiFi devices with small battery capacity, e.g., [1]–[3]. In [1], the authors proposed to reduce the power consumption in idle-listening by lowering the clock-rate in hardware. The authors of [2] found that IEEE 802.11's Power Save Mode (PSM) was not efficient in the presence of competing background traffic, and proposed a scheduling algorithm to resolve it. In [3], the devices enter the sleep mode during the backoff like our scheme, but without controlling the sleep period distribution. While these studies improve the energy efficiency of WiFi

networks, they cannot improve the lifetime, throughput, and fairness performance simultaneously.

## III. SYSTEM MODEL

We first consider a WiFi network with 1 AP and  $N$  devices, where all the devices are associated with the AP. Extensions to general multiple AP scenario are discussed in Section VI.

### A. Energy Model

Consider that device  $n$  will be used for a continuous time duration no shorter than  $T_{target,n}$  before its battery dies out, where  $T_{target,n}$  is called the *target lifetime* of device  $n$ . The lifetime constraint of device  $n$  is given by

$$T_n \geq T_{target,n}, \forall n \in \mathcal{N}, \quad (1)$$

where  $T_n$  is the actual lifetime of device  $n$  and  $\mathcal{N} = \{1, 2, \dots, N\}$  is the set of devices.

The energy cost of each device consists of two parts: the energy spent on the WiFi radio and the base energy consumption in powering the non-WiFi components, such as CPU, screen, etc. Let  $E_{RF,n}$  denote the average energy consumption rate of the WiFi circuit in device  $n$  when turned on, and  $E_{nonRF,n}$  denote the average energy consumption rate of the non-WiFi components in device  $n$ . We assume that, when the WiFi radio is in sleep mode, its energy consumption rate is negligible compared to  $E_{RF,n}$ . For example, measurements showed that the WiFi radio in HTC Tilt mobile phone consumes an average power of 72mW in the sleep mode while it consumes 1120 mW in the awake mode [2]. Let  $B_n$  represent the amount of energy initially stored in the battery, and  $r_n$  the average energy replenishment rate for device  $n$  (either from renewable energy source or from electric power supply), which can be obtained from historical data.<sup>1</sup> Then, the maximum allowable energy consumption rate of the WiFi radio is

$$e_{con,n} = \frac{B_n}{T_{target,n}} + r_n - E_{nonRF,n}, \forall n \in \mathcal{N}. \quad (2)$$

Therefore, the lifetime constraint (1) can be rewritten as the energy consumption constraint as follows

$$e_n \leq e_{con,n}, \forall n \in \mathcal{N}, \quad (3)$$

where  $e_n$  is the actual energy consumption rate of the WiFi radio. In practice, the energy replenishment rate of the solar charger  $r_n$  is usually low such that  $r_n \leq E_{nonRF,n}$ . In order to make sure that the lifetime constraint is feasible, we require that  $e_{con,n} \geq 0$ . Therefore,  $T_{target,n}$  is no greater than  $T_{max,n} \triangleq B_n / (E_{nonRF,n} - r_n)$ , where  $T_{max,n}$  is the maximum feasible lifetime of device  $n$ .<sup>2</sup>

<sup>1</sup>If a device relies only on the battery, we have  $r_n = 0$ .

<sup>2</sup>If a device recharges from electric power supply, then  $r_n > E_{nonRF,n}$  and  $T_{max,n}$  is infinite.

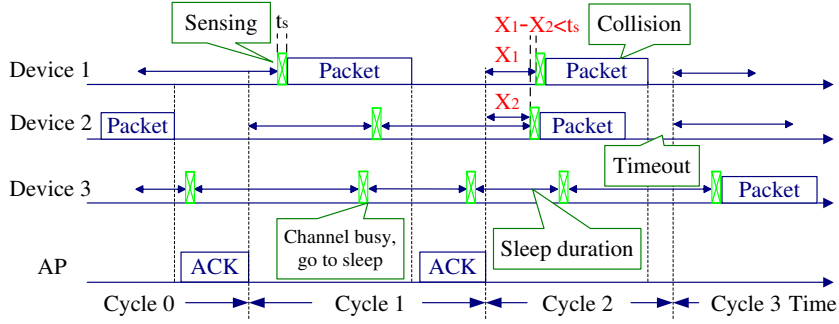


Fig. 1: Illustration of the sleep-wake channel contention scheme.

### B. Sleep-Wake Channel Contention

We now describe our sleep-wake channel contention scheme. Consider the uplink scenario, i.e., from the devices to the AP. When a device wakes up from a sleep mode, it performs carrier sensing immediately if it has a packet to transmit. We use  $t_s$  to represent the carrier sensing time. Since  $t_s$  is very short, we assume that the energy spent on carrier sensing is negligible; see the Remark in Section V-D. If the channel is free, the device transmits the packet; otherwise, the device goes back to sleep immediately. The sleep period for each device  $n \in \mathcal{N}$  is an independent exponentially distributed random period of time with mean  $1/R_n$ , and the control of  $R_n$  will be discussed later.

If two devices begin transmitting within a duration shorter than the carrier sensing time  $t_s$ , they may not be able to detect each other's transmissions and a collision will occur. We assume that the devices are within each other's sensing range.<sup>3</sup> Therefore, each device can detect the transmissions of another device as long as the difference of their wake-up times is larger than  $t_s$ . If the AP successfully receives the packet, it replies with an ACK. After data transmission, the WiFi radio stays awake for an ACK or a timeout. Then it goes back to sleep for another exponentially distributed time duration. The average transmission time of a data packet is denoted by  $L$  and the duration of the ACK is  $t_a$ . Moreover, we assume that when there is a collision, the average collision time plus timeout can be approximated to be  $L + t_a$ . Note that packet collisions may also occur when some devices wake up before the ACK transmission. We omit these collisions in our model to keep the formulated design problem solvable. For similar reasons, we also omit the events that some devices wake up during the timeout [7]. However, they will be considered in our ns-3 simulations in Section VII.

We define a *sleep-wake cycle* as the duration between the end of two successive timeouts or ACKs. Figure 1 illustrates the sleep-wake cycles of three contending devices. After the first ACK, the channel becomes idle and then the one with the smallest remaining sleep period, i.e., Device 1, wakes up and begins transmission. Since no other devices is awake in the preceding and the following period of length  $t_s$ , the transmission is successful, and the AP replies with an ACK. In the next sleep-wake cycle, since Device 1 and 2 wake up

and transmit within a time period shorter than  $t_s$ , it leads to a collision. If the transmitting device does not receive the ACK before the timeout, it goes back to sleep for another exponentially distributed time duration and again contends for the channel to retransmit the packet.

For the downlink scenario, IEEE 802.11 defines a power saving mode (PSM), where the AP periodically broadcasts beacons to notify which devices have pending data packets in the buffer. Each device with pending packets then sends a short packet, called ps-poll packet, to the AP for channel contention. After receiving a ps-poll packet from one winning device, the AP transmits the downlink packets to the device [16]. Note that the channel contention of the downlink case is similar to the uplink case. We focus on the uplink scenario in this paper, and will extend it to the general scenario with both uplink and downlink traffic in our future work.

## IV. PROBLEM FORMULATION

### A. Characterization of Throughput and Energy Consumption

Let  $X_n$  denote the residual sleeping period of device  $n$  when the latest sleep-wake cycle is over. Since the sleeping period of device  $n$  is exponentially distributed with parameter  $R_n$ ,  $X_n$  is also exponentially distributed with parameter  $R_n$ , owing to the memoryless property of exponential distribution.

Let  $\beta_n$  denote the probability that device  $n$  obtains the channel and transmits successfully after a sleep-wake cycle. According to our sleep-wake channel contention scheme, device  $n$  obtains the channel if  $X_i \geq X_n + t_s$  for all  $i \in \mathcal{N}$  and  $i \neq n$ . Therefore,  $\beta_n$  is given by

$$\begin{aligned}
 \beta_n &= \Pr(X_i \geq X_n + t_s, \forall i \neq n) \\
 &\stackrel{(a)}{=} \mathbb{E}[\Pr(X_i \geq X_n + t_s, \forall i \neq n | X_n)] \\
 &\stackrel{(b)}{=} \mathbb{E} \left[ \prod_{i \neq n} \Pr(X_i \geq X_n + t_s | X_n) \right] \\
 &= \int_0^\infty \left[ \prod_{i \neq n} \exp(-R_i(x_n + t_s)) \right] R_n \exp(-R_n x_n) dx_n \\
 &= \frac{R_n \exp(R_n t_s)}{(\sum_i R_i) \exp(\sum_i R_i t_s)}, \tag{4}
 \end{aligned}$$

where (a) is due to  $\Pr[A] = \mathbb{E}[\Pr[A|B]]$ , and (b) is due to the fact that  $X_i$  is independent for different devices.

<sup>3</sup> This assumption is removed in Section VI when studying the case with multiple AP scenarios.

Let  $p_n$  denote the fraction of time that device  $n$  transmits successfully with no collision. Each sleep-wake cycle consists of three parts: transmission/collision, ACK/timeout and idle period. The collision probability is determined by  $\beta_{col} = 1 - \sum_n \beta_n$ , because for any sleep-wake cycle, it includes either a successful transmission or a collision. The idle period is the shortest sleeping period among  $N$  exponentially distributed variables with parameter  $R_i, \forall i \in \mathcal{N}$ . It is easy to show that the idle period is exponentially distributed with parameter  $\sum_i R_i$ . In our sleep-wake channel contention model in Section III-B, the duration of each sleep-wake cycle is *i.i.d.* By the renewal theory in stochastic processes [17],  $p_n$  is given by

$$\begin{aligned} p_n &= \frac{\beta_n \mathbb{E}[\mathbf{trans}]}{(\sum_i \beta_i + \beta_{col})(\mathbb{E}[\mathbf{trans}] + \mathbb{L}[\mathbf{ACK}]) + \mathbb{E}[\mathbf{Idle}]} \\ &= \frac{\beta_n L}{L + t_a + \frac{1}{\sum_i R_i}} \\ &= \frac{R_n \exp(R_n t_s)}{(\frac{L+t_a}{L} \sum_i R_i + \frac{1}{L}) \exp(\sum_i R_i t_s)}, \end{aligned} \quad (5)$$

where  $\mathbb{E}[\mathbf{trans}]$  represents the average packet transmission time  $L$ ,  $\mathbb{L}[\mathbf{ACK}]$  represents the duration of ACK  $t_a$ ,  $\mathbb{E}[\mathbf{Idle}]$  represents the expectation of the idle duration given by  $1/\sum_i R_i$ , and we have used the assumption that the average collision time plus timeout can be approximated to be  $L + t_a$ .

Let  $\bar{D}_n$  be the throughput of device  $n$ . In practice,  $\bar{D}_n$  is proportional to the fraction of successful transmission time  $p_n$ :

$$\bar{D}_n = p_n \alpha_n, \quad (6)$$

where  $\alpha_n$  is a scaling parameter depending on the physical layer signal structure, channel coding and modulation. Note that our control solution does not require knowledge of  $\alpha_n$ .

On the other hand, besides successful transmissions, each device  $n$  may suffer from collisions, which also costs energy. Thus, we denote  $P_n$  as the total fraction of time that the WiFi radio in device  $n$  is turned on, i.e., transmitting or waiting for the ACK. Similar to (5), we can show that

$$P_n = \frac{[1 - \exp(-R_n t_s)] \sum_i R_i + \exp(-R_n t_s) R_n}{\sum_i R_i + \frac{1}{L+t_a}}. \quad (7)$$

The derivation of (7) is provided in [18]. Therefore, the actual energy consumption rate of the WiFi radio is  $e_n = P_n E_{RF,n}$ . Hence, the energy constraints (3) can be rewritten as

$$P_n E_{RF,n} \leq e_{con,n}, \quad \forall n \in \mathcal{N}. \quad (8)$$

Let us define  $b_n \triangleq e_{con,n}/E_{RF,n}$  as the *target energy efficiency* of device  $n$ , then (8) can be rewritten as

$$P_n \leq b_n, \quad \forall n \in \mathcal{N}. \quad (9)$$

Note that this constraint is always satisfied when  $b_n \geq 1$ .

### B. Problem Statement

Suppose that each device  $n$  is associated with a proportional-fair utility function  $\ln(\bar{D}_n)$  [19]. Our goal is to design the

average sleep period  $R_i$  for  $i \in \mathcal{N}$  to maximize the total utility. This design problem can be formulated as

$$\max_{\bar{R} > 0} \sum_{i \in \mathcal{N}} \ln(\bar{D}_i), \quad (10)$$

subject to the per-device energy constraint (9).

Substituting (5) and (7) into (9) and (10), this problem is rewritten as the following problem: (Problem **A**)

$$\begin{aligned} \max_{R_i > 0} \quad & \sum_i \ln R_i - N \ln \left( \sum_i R_i + \frac{1}{L+t_a} \right) \\ & - (N-1) \sum_i R_i t_s + N \ln \left( \frac{L}{L+t_a} \right) + \sum_i \ln \alpha_i \\ \text{s.t.} \quad & \frac{[1 - \exp(-R_n t_s)] \sum_i R_i + \exp(-R_n t_s) R_n}{\sum_i R_i + \frac{1}{L+t_a}} \leq b_n, \forall n. \end{aligned}$$

### V. PROBLEM SOLUTION

Problem **A** is a non-convex optimization problem. We first consider a relaxed version of Problem **A**, i.e.,

$$\begin{aligned} \max_{R_i > 0} \quad & \sum_i \ln R_i - N \ln \left( \sum_i R_i + \frac{1}{L+t_a} \right) \\ & - (N-1) \sum_i R_i t_s + N \ln \left( \frac{L}{L+t_a} \right) + \sum_i \ln \alpha_i \end{aligned} \quad (11)$$

$$\text{s.t. } R_n \leq b_n \left( \sum_i R_i + \frac{1}{L+t_a} \right), \forall n. \quad (12)$$

The only difference between the above problem and Problem **A** is the constraints. Since  $R_n \leq [1 - \exp(-R_n t_s)] \sum_i R_i + \exp(-R_n t_s) R_n, \forall n$ , the constraints of (11) are looser than those of Problem **A**. Hence, (11) serves as a performance upper bound of Problem **A**.

In the sequel, we solve (11) in two cases:  $\sum_i b_i \geq 1$  and  $\sum_i b_i < 1$ , and show in Theorem 1 that the obtained solution is asymptotically optimal for the original Problem **A** as  $\frac{t_s}{L+t_a} \rightarrow 0$ .

#### A. The Case of $\sum_i b_i \geq 1$

In this case, we provide an approximate solution to Problem (11) in two steps:

- (1) Define an auxiliary variable  $y = \sum_i R_i$ . Given that  $y > 0$ ,  $R_i$  is determined by the following approximate problem:

$$\max_{R_i > 0} \sum_i \ln R_i - N \ln \left( y + \frac{1}{L+t_a} \right) \quad (13)$$

$$- (N-1) y t_s + N \ln \left( \frac{L}{L+t_a} \right) + \sum_i \ln \alpha_i,$$

$$\text{s.t. } R_n \leq b_n y, \quad \forall n \in \mathcal{N} \quad (14)$$

$$\sum_i R_i = y. \quad (15)$$

In [18], we show that the optimum solution to (13) is determined by

$$R_n = \min\{b_n, c^*\} y, \quad (16)$$

where  $c^*$  satisfies

$$\sum_i \min\{b_i, c^*\} = 1. \quad (17)$$

(2) Substituting (16) and (17) into (13), we get

$$\begin{aligned} & \max_{y>0} N \ln y - N \ln \left( y + \frac{1}{L+t_a} \right) - (N-1)yt_s \\ & + N \ln \left( \frac{L}{L+t_a} \right) + \sum_i \ln \alpha_i + \sum_i \ln[\min\{b_i, c^*\}], \end{aligned} \quad (18)$$

which is an one-dimensional convex optimization problem of  $y$ . Taking the derivative of the objective function and setting it to be zero, yields

$$y^* = -\frac{1}{2(L+t_a)} + \frac{\sqrt{1 + \frac{4N(L+t_a)}{(N-1)t_s}}}{2(L+t_a)}. \quad (19)$$

Substituting  $y = y^*$  into (16), the sleep period parameter  $\{R_i\}$  is derived.

Note that Problem (13) has more restricted energy constraints than Problem (11). Hence our solution (16), (17), and (19) is feasible for Problem (11). Since (16), (17), and (19) achieves an objective value given by (18), the optimum objective value of Problem (11) is lower bounded by (18).

#### B. The Case of $\sum_i b_i < 1$

Substituting the energy constraints (12) into (11), the optimum value of (11) is upper bounded by

$$\begin{aligned} & \max_{R_i>0} \sum_i \ln b_i - (N-1) \sum_i R_i t_s + \sum_i \ln \left( \frac{L\alpha_i}{L+t_a} \right), \quad (20) \\ \text{s.t. } & R_n \leq b_n \left( \sum_i R_i + \frac{1}{L+t_a} \right), \forall n. \end{aligned}$$

In [18], we prove that the optimum solution to (20) satisfies

$$R_n = b_n \left( \sum_i R_i + \frac{1}{L+t_a} \right), \quad \forall n. \quad (21)$$

Moreover, if (21) holds, the optimum values of (11) and (20) are identical. If  $\sum_i b_i < 1$ , the linear system (21) has a unique solution, which is given by

$$R_n = \frac{b_n}{(L+t_a)(1-\sum_i b_i)}, \quad \forall n. \quad (22)$$

By this, we have obtained the optimum solution to Problem (11) for  $\sum_i b_i < 1$ .

#### C. Implementation Procedure

We first express the solutions for  $\sum_i b_i \geq 1$  and  $\sum_i b_i < 1$  in a unified form. For  $\sum_i b_i < 1$ , the solution (22) can also be expressed by (16), where  $c^*$  and  $y^*$  are determined as

$$c^* = 1, \quad y^* = \frac{1}{(L+t_a)(1-\sum_i b_i)}. \quad (23)$$

Our transmission control scheme is described as follows:

#### **Life-Add: Lifetime Adjustable Design for WiFi Networks**

- (1) The AP gathers the system parameters  $N$ ,  $L$ ,  $t_a$  and  $t_s$ , and the target energy efficiency parameter  $b_i$  from the devices.
- (2) The AP checks if  $\sum_i b_i \geq 1$  or  $\sum_i b_i < 1$ . If  $\sum_i b_i \geq 1$ , the AP computes  $c^*$  and  $y^*$  according to (16) and (17); otherwise, it computes  $c^*$  and  $y^*$  according to (23).

(3) The AP broadcasts  $c^*$  and  $y^*$  to the devices by piggybacking them on its advertisement beacons.

(4) On hearing  $c^*$  and  $y^*$ , the device  $n$  computes  $R_n$  by (16), and then utilizes  $R_n$  to generate its sleep period.

#### D. Performance Analysis

The performance of Life-Add is characterized by the following theorem:

**Theorem 1:** Our solution (16), (17), (19) and (23) is asymptotically optimal for Problem A as  $\frac{t_s}{L+t_a} \rightarrow 0$ .

*Proof:* Here, we provide a proof sketch consisting of two steps. We refer to a more detailed proof in [18].

In Step One, we show that our solution is asymptotically optimal for Problem (11) as  $\frac{t_s}{L+t_a} \rightarrow 0$ . If  $\sum_i b_i < 1$ , we have derived the exact solution to Problem (11) in Section V-B. If  $\sum_i b_i \geq 1$ , the optimum objective value of Problem (11) is lower bounded by (18), which is achieved by our solution (16), (17), and (19). On the other hand, we shown in [18] that the optimum objective value of Problem (11) is upper bounded by

$$N \ln \left( \frac{L}{L+t_a} \right) + \sum_i \ln \alpha_i + \sum_i \ln[\min\{b_i, c^*\}]. \quad (24)$$

Moreover, the gap between the lower bound (18) and upper bound (24) tends to 0 as  $\frac{t_s}{L+t_a} \rightarrow 0$ . Therefore, our solution (16), (17), and (19) is asymptotically optimal for Problem (11) as  $\frac{t_s}{L+t_a} \rightarrow 0$  if  $\sum_i b_i \geq 1$ .

In Step Two, we show that our solution satisfies  $R_n t_s \rightarrow 0$  as  $\frac{t_s}{L+t_a} \rightarrow 0$  for both  $\sum_i b_i \geq 1$  and  $\sum_i b_i < 1$ . By this, Problem (11) and Problem A tends to be the same problem as  $\frac{t_s}{L+t_a} \rightarrow 0$ . Therefore, our asymptotically optimal solution to Problem (11) is also asymptotically optimal for Problem A. ■

**Remark:** It is worth pointing out that the condition  $\frac{t_s}{L+t_a} \approx 0$  is satisfied in practical WiFi networks. For example, in IEEE 802.11b WiFi standard, the carrier sensing time is  $t_s = 4\mu\text{s}$  [20], and the transmission time of the data packet and ACK varies from  $511\mu\text{s}$  to  $1573\mu\text{s}$  for a packet with 0 to 1460 bytes of payload [21]. Hence,  $\frac{t_s}{L+t_a} \leq 0.00783$ .

#### VI. GENERAL WiFi NETWORKS WITH MULTIPLE APs

We now extend Life-Add to more general scenarios with multiple APs, where some clients and/or APs cannot detect the existence of each other because of the large network size. The key challenge here is the complicated interference environment, which makes the transmission control problem very difficult.

One such example is the asymmetric interference scenario shown in Fig. 2, which is also known as the *near-far effect* in wireless communications. In this example, Device 1 is associated with AP 1 and Device 2 is associated with AP 2. Since AP 1 is out of the interference range of Device 2, it can successfully receive the packets from Device 1. However, the transmissions of Device 2 are seriously interfered by Device 1. In the IEEE 802.11 WiFi designs, Device 1 will keep the minimum backoff since its transmissions are always successful, while Device 2 has high backoff time due to packet collisions at AP 2. This significantly reduces the throughput of Device 2 and results in unfairness between the two devices [22]. Note

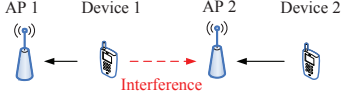


Fig. 2: An illustrating example of the near-far effect.

that the RTS-CTS mechanism has large overhead, thus it cannot improve the throughput performance significantly.

Another challenge of the multiple AP scenario is the *hidden terminal problem*, where a packet collision may occur at a AP even if the transmission starting time of two packets are more than  $t_s$  apart, because the two devices cannot detect each other. Therefore, the collision probability  $\beta_{col}$  is under-estimated.

We now extend Life-Add to the multi-AP scenario, and use a heuristic design to handle these two challenges. In order to compute the  $c^*$  and  $y^*$  in multi-AP networks, each AP overhears the reports of  $b_i$  from the devices within its communication range, including the devices that are associated with other APs. Similar overhearing procedure is supported in IEEE 802.11 MAC [23]. For example, to perform virtual carrier sensing, each device overhears the packets from nearby devices to get the packet length information contained in the packet header. In addition, each device receives the beacons broadcasting  $c^*$  and  $y^*$  from all the APs within its communication range.

To resolve the near-far effect issue, Life-Add utilizes the following device collaboration scheme: each device computes the  $R_n$ s suggested by nearby APs, and chooses to use the smallest  $R_n$  value, which corresponds to the largest average sleep duration. This helps to improve fairness among the devices. For example, for the network shown in Fig. 2, Life-Add will compute the backoff time of Device 1, by considering channel contentions at both AP 1 and AP 2. This is as if nearby APs tell Device 1 how many devices' transmissions it interferes with at each AP (in our example, Device 1 interferes with Device 2 at AP 2). Therefore, Device 1 transmits conservatively to provide more channel access opportunities to Device 2 compared to IEEE 802.11.

For the hidden terminal problem, Life-Add employs the following congestion control scheme similar to that of IEEE 802.11 [23]: Each device increases its average sleep period  $1/R'_n$  when it undergoes a timeout, and reduces  $1/R'_n$  when it receives an ACK, where  $R'_n$  is the actual sleep-wake parameter adopted by the device. The sleep-wake parameter  $R_n$  assigned by the APs is used as an upper bound of the actual sleep-wake parameter  $R'_n$  to ensure the lifetime constraint.

Life-Add for the multiple AP scenario is as follows:

- (1) Each AP gathers the system parameters  $L$ ,  $t_a$ ,  $t_s$  and  $N$ , where  $N$  is the number of devices within its communication range (including those associated with other APs). Each AP also collects  $b_i$  from all these devices.
- (2) Each AP checks whether  $\sum_i b_i \geq 1$  or  $\sum_i b_i < 1$ . If  $\sum_i b_i \geq 1$ , the AP computes  $c^*$  and  $y^*$  by (16) and (17); otherwise, it computes  $c^*$  and  $y^*$  by (23).
- (3) Each AP broadcasts  $c^*$  and  $y^*$  to the devices within its communication range by piggybacking them on its advertisement beacons.

- (4) Each device  $n$  may receive multiple pairs of  $(c^*, y^*)$  from nearby APs within its communication range. For each  $(c^*, y^*)$ , it computes the corresponding  $R_n$  using (16), and chooses to use the smallest  $R_n$  value.
- (5) Each device computes its *Congestion Control Factor*  $F_n$ . Initially, the value of  $F_n$  is 1. Each time the device undergoes a timeout, it doubles its  $F_n$  (up to a maximum value of 32). Each time the device receives an ACK, its  $F_n$  is reset to 1. The actual sleep-wake parameter  $R'_n$  is computed by  $R'_n = R_n/F_n$ . Finally, the sleep period is generated as an exponentially distributed random variable with mean  $1/R'_n$ .

Life-Add differs from IEEE 802.11 MAC in the following aspects: (1) In Life-Add, a device transmits immediately after it senses the channel to be idle. However, in IEEE 802.11 MAC, the device keeps sensing the channel for an additional backoff time, which consumes much energy. This is known as the “idle-listening” problem in the literature [1]. (2) By using  $b_n$  to compute  $R_n$ , Life-Add takes the energy supply and lifetime requirement of the devices into consideration. (3) Life-Add intends to improve the throughput and fairness performance by carefully controlling the transmission parameters and introducing the device collaboration scheme.

We finally note that a simpler version of Life-Add can be derived by ignoring the lifetime constraints and simply choosing  $b_n > 1$ . This corresponds to the best throughput performance of Life-Add, since the devices do not need to sacrifice the throughput to satisfy their lifetime constraints. This case will be considered in our simulations in Section VII-B.

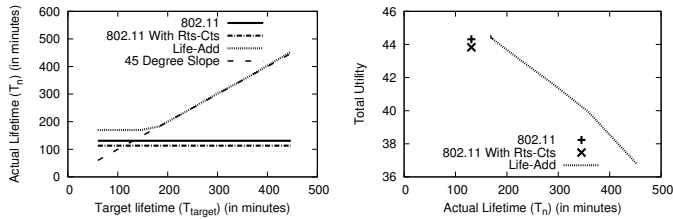
## VII. SIMULATION RESULTS

We now provide ns-3 [24] simulation results to evaluate the performance of Life-Add, where its physical layer is the same as IEEE 802.11b. For the purpose of comparison, we also simulate IEEE 802.11b MAC as well as IEEE 802.11b MAC with RTS-CTS. All the simulation results are averaged over 5-10 system realizations. We evaluate our system under UDP saturation conditions such that the devices always have UDP data to send. The packet length varies according to the measurements in [25].

### A. Single AP Scenario

We set up a WiFi network with 1 AP and 3 associated devices in a field of size 50m  $\times$  50m. The sensing threshold is set to -100 dBm resulting in a sensing range of approximate 110m. Therefore, all the devices can hear each other. The AP is assumed to be connected to the wall power while each device is equipped with a battery and a solar panel that provides renewable energy charging. We assume that the devices are HTC Tilt smartphones, where if the WiFi radio is awake, the device consumes<sup>4</sup> 1435 mW, and if the WiFi radio is in sleep

<sup>4</sup>It was measured in [2] that the energy consumption rate of the WiFi radio of HTC Tilt 8900 series in awake mode is 1120 mW and the base energy consumption rate is within 155mW 475 mW (an average of 315 mW) for a total consumption of 1435 mW. When the radio is in sleep mode, the total consumption is 315 mW + 72 mW = 387 mW.



(a) Average actual lifetime vs. target lifetime (b) Total Utility vs. actual lifetime

Fig. 3: Simulation results for single AP networks with homogeneous devices.

TABLE II. Parameters of the heterogeneous devices.

Node	Initial Battery Level (in mAh)	Recharging Rate (in mW)	Target lifetime (in mins.)
$N_1$	200	187	18,36, ..., 180
$N_2$	100	90	9,18, ..., 90
$N_3$	66.6	67	6,12, ..., 60

mode, the energy consumption is 387 mW. Each device has a battery with a capacity of 1200 mAh.

1) *Homogeneous Devices*: We consider a homogeneous scenario where all devices have the same initial battery level (300 mAh) and recharging rate (160 mW). Fig. 3(a) shows the average actual lifetime of the three designs with varying target lifetime  $T_{target}$ . The actual lifetime of Life-Add increases with respect to the target lifetime, while the actual lifetimes of the other two designs stay constant. This figure shows that the actual lifetime of Life-Add is adjustable and satisfies the lifetime constraint (1). We note that even the minimum lifetime value of Life-Add is longer than that of the IEEE 802.11 MACs, because there is no idle-listening in Life-Add.

Fig. 3(b) shows how Life-Add realizes lifetime extension by trading the total utility. The utility of a device is determined by  $U(x) = \ln(x)$ , if the average throughput of the device is  $x$  kbps. Life-Add can achieve tunable lifetime by adjusting the average sleeping-backoff durations of the devices. However, for the IEEE 802.11 MACs, it is not possible to adjust the utility and the actual lifetime, and thus, the devices always have fixed utility and lifetime.

2) *Heterogeneous Devices*: We now explore the performance of Life-Add for a heterogeneous scenario. The initial battery level  $B_n$ , energy recharging rate  $r_n$ , and target lifetime values  $T_{target,n}$  of the three devices are provided in Table II, where the varying step sizes of  $T_{target,n}$  for the three devices are different. Fig. 4 shows the tradeoff between the throughput and actual lifetime of all three devices. We can see that Life-Add can achieve higher throughput than the IEEE 802.11 MACs for each device, which also implies a larger total utility.

### B. Multiple AP scenario with Heterogeneous Devices

1) *With Varying Number of Devices*: We setup a field of  $300m \times 300m$ , in which we randomly deploy 4 APs and varying number of devices. Each device is associated with the AP that has the strongest signal, and the device only sends data packets to its associated AP. The devices have heterogeneous

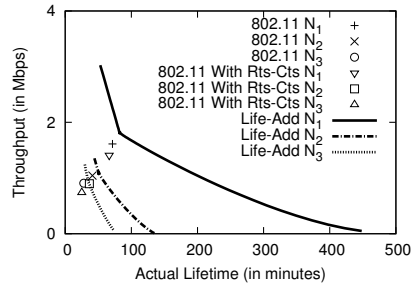
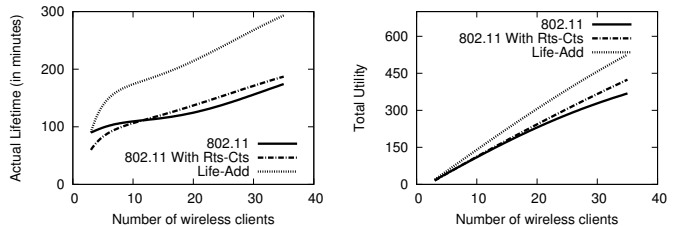


Fig. 4: Simulation results for single AP networks with heterogeneous devices.



(a) Average actual lifetime vs. number of devices (b) Total utility vs. number of devices

Fig. 5: Simulation results for multi AP networks with heterogeneous devices: varying number of devices.

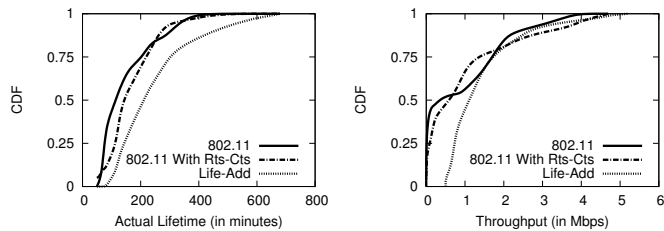
energy supplies. In particular, 1/3 devices only have access to the energy stored in their batteries. Another 1/3 devices have access to both battery energy and renewable energy from solar panel. The last 1/3 devices are connected to the wall power that provides high energy recharging rate. The initial battery levels of all the devices are randomly generated by a uniform distribution within the range of [200, 1000] mAh. The lifetime constraints are ignored here by choosing  $b_i > 1$ , which achieves the best possible throughput performance.

Figure 5(a) shows the average lifetime performance versus the number of devices. Life-Add prolongs the average lifetime because there is no idle-listening. Figure 5(b) shows the total utility provided by these three designs. One can see that Life-Add achieves a larger proportionally fair utility.

2) *CDF*: Now we consider a field of  $500m \times 500m$ , in which we randomly deploy 4 APs and 30 devices. The energy supply of the devices are set as in Section VII-B1. Figure 6(a) shows the CDF (cumulative distribution function) of the actual lifetime of the devices. The figure only plots the CDF for the first 20 devices since the last 10 devices have infinite lifetime due to the high recharging rate provided by the wall power. Figure 6(b) shows the CDF of the throughput of the individual devices (for all 30 devices). The average lifetime, average throughput, Jain's fairness index [6], and successful transmission probability for this simulation scenario is provided in Table I. Life-Add has better throughput performance because IEEE 802.11 has too many collisions and the RTS-CTS mechanism is not efficient. It is relative easier to understand the lifetime and fairness improvement.

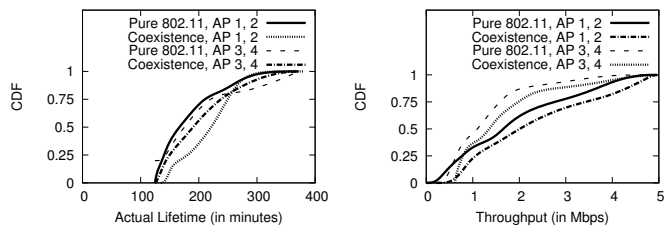
3) *Coexistence with IEEE 802.11*: Finally, we explore the coexistence between IEEE 802.11 and Life-Add. We set up





(a) CDF of the actual lifetime of the devices (b) CDF of the throughput of the devices

Fig. 6: Simulation results for multi AP networks with heterogeneous devices: CDF performance.



(a) CDF of the actual lifetime of the devices (b) CDF of the throughput of the devices

Fig. 7: Simulation results for the coexistence of Life-Add and IEEE 802.11.

the network as explained in Section VII-B2. However, this time AP 1 and 2 (randomly chosen) switch from IEEE 802.11 (without RTS-CTS) to Life-Add while AP 3 and 4 stick to IEEE 802.11. Each device is associated with the AP with the strongest signal, which then determines the design used by the device. Fig. 7 shows the lifetime and throughput CDFs of the devices associated with AP 1-2 and AP 3-4, before and after AP 1-2 switch to Life-Add. One can observe that, after AP 1-2 switch to Life-Add, the lifetime distribution of the devices using Life-Add is significantly improved, because there is no idle-listening in Life-Add. The lifetime distribution of the devices stick to IEEE 802.11 does not change much. On the other hand, the throughput distributions for both groups of devices are improved. Therefore, upgrading IEEE 802.11 to Life-Add is beneficial for both the devices switching to Life-Add and those sticking to IEEE 802.11, which provides the incentive to employ Life-Add.

### VIII. CONCLUSION

In this paper, we have proposed Life-Add, a life-time adjustable design for WiFi networks. By using a sleep-wake channel contention scheme, Life-Add can resolve the high energy consumption of the current WiFi design. Then, by controlling the sleep period parameters carefully and introducing collaboration among the devices, Life-Add has a small packet collision probability, which implies high throughput. Finally, by providing high priority to the low throughput devices, Life-Add improves the fairness performance. Our simulation results show that Life-Add has better lifetime, throughput, and fairness performance than IEEE 802.11b, and coexists harmoniously with IEEE 802.11b. For future work, we are evaluating how

Life-Add behaves under TCP traffic and in the presence of downlink traffic. Its hardware implementation is currently under preparation.

### REFERENCES

- [1] X. Zhang and K. G. Shin, "E-MiLi: Energy-minimizing idle listening in wireless networks," in *MobiCom*, 2011, pp. 205–216.
- [2] E. Rozner, V. Navda, R. Ramjee, and S. K. Rayanchu, "NAPman: Network-assisted power management for WiFi devices," in *MobiSys*, 2010, pp. 91–106.
- [3] V. Baiamonte and C.-F. Chiasserini, "Saving energy during channel contention in 802.11 WLANs," *Mob. Netw. Appl.*, vol. 11, no. 2, pp. 287–296, 2006.
- [4] Firenew Technology Company, "Solar mobile charger," <http://www.firenew.com.cn/product.asp?ncclassid=248&typeid=71>.
- [5] Mophie Inc., "External USB portable battery charger," <http://www.mophie.com/category-s/53.htm>.
- [6] R. Jain, *The Art of Computer Systems Performance Analysis*. John Wiley & Sons, 1991.
- [7] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.
- [8] L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 3, pp. 960–972, Jun. 2010.
- [9] M. J. Neely, "Energy optimal control for time-varying wireless networks," *IEEE Trans. Inf. Theory*, vol. 52, no. 7, pp. 2915–2934, 2006.
- [10] L. Lin, X. Lin, and N. B. Shroff, "Low-complexity and distributed energy minimization in multihop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 2, pp. 501–514, 2010.
- [11] S. Chen, P. Sinha, N. Shroff, and C. Joo, "A simple asymptotically optimal energy allocation and routing scheme in rechargeable sensor networks," in *IEEE INFOCOM 2012*, Mar. 2012, pp. 379–387.
- [12] L. Huang and M. J. Neely, "Utility optimal scheduling in energy harvesting networks," in *MobiHoc*, 2011.
- [13] J. Kim, X. Lin, N. Shroff, and P. Sinha, "Minimizing delay and maximizing lifetime for wireless sensor networks with anycast," *IEEE/ACM Trans. Netw.*, vol. 18, no. 2, pp. 515–528, Apr. 2010.
- [14] Y. Z. Zhao, C. Miao, M. Ma, J. B. Zhang, and C. Leung, "A survey and projection on medium access control protocols for wireless sensor networks," *ACM Comput. Surv.*, vol. 45, no. 1, pp. 7:1–7:37, 2012.
- [15] A. Bachir, M. Dohler, T. Watteyne, and K. Leung, "MAC essentials for wireless sensor networks," *IEEE Commun. Surv. Tutorials*, vol. 12, no. 2, pp. 222–248, Quarter 2010.
- [16] 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, 2008.
- [17] R. G. Gallager, *Discrete Stochastic Processes*. Boston: Kluwer Academic Publishers, 1996.
- [18] S. Chen, T. Bansal, Y. Sun, P. Sinha, and N. B. Shroff, "Life-Add: Lifetime Adjustable design for WiFi networks with heterogeneous energy supplies," 2013, Technical Report, Depts of ECE and CSE, Ohio State University. [Online]. Available: <http://www.ece.osu.edu/~chens/wiopt13.pdf>
- [19] F. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, 1997.
- [20] E. Magistretti, K. K. Chintalapudi, B. Radunovic, and R. Ramjee, "WiFi-Nano: reclaiming WiFi efficiency through 800 ns slots," in *MOBICOM*, 2011.
- [21] M. Gast, "When Is 54 Not Equal to 54? A Look at 802.11a, b, and g Throughput," [http://www.oreillynet.com/pub/a/wireless/2003/08/08/wireless\\_throughput.html?page=1](http://www.oreillynet.com/pub/a/wireless/2003/08/08/wireless_throughput.html?page=1).
- [22] C. Hua and R. Zheng, "Starvation modeling and identification in dense 802.11 wireless community networks," in *IEEE INFOCOM 2008*, 2008, pp. 1022–1030.
- [23] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, Institute of Electrical and Electronics Engineers, 2007, <http://standards.ieee.org/about/get/802/802.11.html>.
- [24] "ns-3," <http://www.nsnam.org/>.
- [25] R. Sinha, C. Papadopoulos, and J. Heidemann, "Internet packet size distributions: Some observations," *USC/Information Sciences Inst. [Online]*. Available: <http://netweb.usc.edu/~rsinha/pkt-sizes>, 2007.