Just-in-time WLANs: On-demand Interference-managed WLAN Infrastructures

Kimin Lee[†], Yeonkeun Kim[‡], Seokhyun Kim[†], Jinwoo Shin[†], Seungwon Shin[‡] and Song Chong[†]

***Abstract*—In the past years, the centralized management of dense wireless local area networks has been emerged as a powerful paradigm for improving energy efficiency as well as avoiding severe interference. In this paper, we study the joint optimization on power-operation modes in access points (APs), channel selections and user-AP associations for improving energy efficiency and avoiding interference without sacrificing users' demands. To this end, we first formulate it as a mixed-integer programming using the popular Lyapunov approach, but it turns out to be computationally intractable, i.e., NP-hard. To address the issue, we propose a polynomial-time approximation algorithm and prove that it achieves a constant-factor approximation guarantee under mild assumptions. The main novelty underlying our algorithm design is based on a linear programming relaxation combining with two different greedy rounding schemes, where each achieves a constant-factor approximation in different regimes of parameters. We verify the performance of the proposed algorithm via extensive simulations and also demonstrate its practicability by implementing it at commercial APs using a Software-defined Networking framework, which shows that it reduces the wasted energy significantly while maintaining even higher throughput.**

I. INTRODUCTION

To serve users' demands for a high bandwidth and reliability, wireless local area networks (WLANs) that are composed of a high-density of access points (APs) have been deployed rapidly in the past years. Although a dense WLAN can serve high traffic demands, it causes two critical issues: wasted energy and severe interference. First, according to [1], the peak traffic demand rarely occurs. Actually, a small fraction of APs is utilized during the day and most ones remain idle during nights and weekends. Hence, the wastage of energy occurs since the majority of APs is always on. Second, because of a lack of non-overlapping channels, the high-density APs inevitably cause the high interference, which has been evidenced to degrade the performance of WLANs [2], [3].

To address the issues, commercial commercial vendors [4], [5] and researchers have actively adopted the centralized controllers [1], [6]–[10]. For example, the joint optimization of AP operation modes (on/off) and AP-user associations in order to reduce the energy waste was studied [1], [6], [7], which show that energy can be saved significantly by reducing static power consumption. In addition, to avoid the severe interference, it has been shown [8]–[10] that the interference can be reduced significantly using 'smart' channel selections. However, to the best of our knowledge, no work is known to study both issues of power consumption and interference together. The two objectives are not mutually exclusive since AP operation mode control can be used not only for the energy saving, but also for the interference alleviation. Motivated by this, in order to reduce the energy consumption and avoid the interference simultaneously, we study a problem of jointly controlling AP operation modes, channel selections and user-AP associations without sacrificing users' demands.

Our contribution. To this end, we first formulate a long-term objective minimizing the average power consumption of APs under constraints in interference and users' demands. To optimize this long-term objective, we formulate a short-term mixed integer programming, say **SMIP**, using the popular Lyapunov approach [11], where computing its optimal solution at each time instance leads to the optimality of this long-term objective. However, it turns out to be computationally intractable (i.e., NP-hard) to solve exactly. As the main contribution of this paper, we develop a constant-factor approximation algorithm for **SMIP** and demonstrate its performance through extensive simulations and testbeds in the real environment, where more details are explained in what follows.

In order to design an approximation algorithm for **SMIP**, we first relax its integral constraints and design a linear programming, say **SLP**. Using an optimal solution of **SLP**, we generate two feasible candidate solutions of **SMIP** from two greedy rounding schemes and choose a better one with respect to the objective of **SMIP**. The proposed rounding schemes are motivated by those used for solving the maximum weighted independent set (MWIS) [12] and the minimum set covering (MSC) [13]. Here, our main intuition is that **SMIP** becomes similar to MWIS and MSC depending on regimens of parameters. This allows us to establish constant-factor approximation guarantees that do not scale with the number of users and APs.

We demonstrate that the proposed algorithm can optimize the long-term objective better than other heuristics via extensive simulations. In addition, we implement a prototype of our algorithm on a Software-defined Networking (SDN) framework and verify our proposed algorithms with this prototype. Since SDN techniques enable us to manage a network in a centralized way and allow us to control a network dynamically, we believe that we can leverage SDN in realizing our ideas. Specifically, SDN provides a global view of network, and thus SDN can achieve a global optimization for a network. Moreover, the programmability of an SDN enables a network-side AP association mechanism rather than a user-side so that the network has a control on the association mechanism. Along with this feature, a decoupled control plane, which enables a dynamic network control, supports the seamless handover for mobile users.

Related work. There have been extensive works to study the

†: Department of Electrical Engineering, KAIST, Korea. Email: {kmlee, sh.kim}@netsys.kaist.ac.kr, jinwoos@kaist.ac.kr, songchong@kaist.edu ‡: School of Computing, KAIST, Korea. Email: {yeonk, claude}@kaist.ac.kr

management of WLANs in the past years. Some researchers [1], [6], [7] developed energy-efficient management schemes. The authors in [7] formulated the energy saving problem as a linear programming to solve it. A threshold-based algorithm [1], [6] was proposed to reduce energy consumption of APs without adversely impacting users' demands in the network. Although these works suggest to save energy by reducing static power consumption, their schemes might suffer from severe wireless interference. On the other hand, there have been also studied about interference management schemes, e.g., transmission power control [2], AP placement optimization [14], and channel selection [3], [8], [15]. However, these works do not take energy saving into account. The authors in [9], [10] consider both interference and energy consumption, but the complexity of the proposed algorithm is quite high (i.e., not polynomial-time) and they control AP placements while we do AP operation modes given placements.

Recently, several studies have leveraged an SDN in WLANs to achieve their designed managements. Odin [16] is an SDN framework to provide programming abstraction of networks and seamless handovers. Yap *et al.* [17] have proposed OpenFlow WiFi APs and WiMax base stations to ease heterogeneous networks. Riggio *et al.* [6] have presented a WLAN management solution for energy saving and mobility on an SDN framework.

II. Preliminaries

A. Network Model

Network interference graph. We model interference among APs by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each node corresponds to an AP and an edge $(u, v) \in \mathcal{E}$ is present if there is an overlap in the interference region of the basic service set created by AP u and v. Hence, a set of non-interfering APs forms an independent set in $\mathcal{G}$. In addition, we let $\mathcal{N}_{\mathcal{G}}(v)$ denote the set of AP v's neighbors in the interference graph $\mathcal{G}$, and $\Delta = \max_{v \in \mathcal{V}} |\mathcal{N}_{\mathcal{G}}(v)|$, i.e., the maximum degree of $\mathcal{G}$, where we assume that Δ does not scale with $|\mathcal{V}|$, i.e., $\Delta = O(1)$.

Control. We consider a centralized management system such that every AP is connected to the central controller. The controller has three functionalities as follows:

- *Operation mode control.* The controller turns on or off APs. When it is unnecessary to turn on some set of APs, the controller turns off them in order to reduce the energy consumption and avoid the interference. We denote an active mode indicator of AP v at time t by $y_v(t) \in \{0, 1\}$, which is 1 if v is on and 0 otherwise.
- *Channel control.* To avoid the interference, the controller selects the operating channel such that no two adjacent APs have to use the same channel. Let $\mathcal{C}$ denote the set of all possible non-overlapping channels available in the underlying PHY layer. We denote an operating channel of AP v at time t by $c_v(t) \in \mathcal{C}$. We assume that $|\mathcal{C}| = O(1)$.
- *Association control.* When a set of active APs changes, users may need to be associated with another AP. Therefore, the controller has to decide the association of users to APs, where we let $\mathcal{I}$ denote the set of users. We denote a connection between user $i \in \mathcal{I}$ and AP v at time t by $x_{iv}(t) \in \{0, 1\}$, which is 1 if i is associated with v and 0 otherwise.

Service rate. Our focus is on downlink communication, i.e., from APs to users, since our goal is to reduce the power consumption of APs. We denote an achievable rate of user i from AP v at time t by $r_{iv}(t)$, where we define $S_v(t) = \{i \in \mathcal{I} | r_{iv}(t) > 0\}$ and $L_i(t) = \{v \in \mathcal{V} | r_{iv}(t) > 0\}$ for 'active' users and APs at time t. In addition, we denote the maximum number of users covered by a single AP at time t by $\Omega(t) = \max_{v \in \mathcal{V}} |S_v(t)|$. Similarly, we define $\Lambda(t) = \max_{i \in \mathcal{I}} |L_i(t)|$. Notice that $\Omega(t)$, $\Lambda(t)$ often do not scale with $|\mathcal{I}|$ and $|\mathcal{V}|$ in many practical scenarios, i.e., $\Omega(t), \Lambda(t) = O(1)$.

We assume that $r_{iv}(t)$ is given at time t and constant during each decision period, i.e., static channels are assumed during each decision period. Since each AP generally serves more than one user, users assigned to same AP have to share resources such as frequencies and time slots. Therefore, each user can be served only in a fraction of the achievable rate $r_{iv}(t)$. We denote a fraction of resource that AP v serves user i at time t by $\theta_{iv}(x, t)$, which is expressed as $\frac{x_{iv}(t)}{\sum_{j \in \mathcal{I}} x_{jv}(t)}$[1]. One can note that we assume an equal-time scheduling such that the load on the AP is proportional to the total number of users assigned to it. Note that it can achieve system-wide proportional fairness [18]. Then, the overall service rate of user i from AP v at time t, denoted by by $D_{iv}(x, t)$, becomes $r_{iv}(t)\theta_{iv}(x, t)$. We assume that $D_{iv}(x, t)$ has an upper bound, i.e., $D_{iv}(x, t) \leq D_{\max}$.

Queueing and power consumption. We denote a traffic arrival rate of user i at time t by $A_i(t)$, where we assume that there exists an upper bound on it, i.e., $A_i(t) \leq A_{\max}$. Because $\sum_{v \in \mathcal{V}} D_{iv}(x, t)$ corresponds to a departure rate of user i, the queue dynamics is expressed as

$$q_i(t+1) = \left[q_i(t) - \sum_{v \in \mathcal{V}} D_{iv}(x, t) \right]^+ + A_i(t). \quad (1)$$

where $[x]^+ = \max(x, 0)$ and $q_i(t)$ is the queue-size of user i at time t. The power consumption of APs is related to the specific operation mode (e.g., idle, RX or TX) [19]. Nevertheless, since our focus is on 'sleeping' idle APs to reduce it, we simply write the power consumption of AP v at time t as $p_v y_v(t)$, where p_v is a static power consumption to turn on AP v. We denote the maximum and minimum power consumption by $p_{\max}$ and $p_{\min}$, respectively. Note that $\frac{p_{\max}}{p_{\min}}$ is constant, i.e., $\frac{p_{\max}}{p_{\min}} = O(1)$.

B. Performance Metric

In order to minimize power consumption of APs taking into account interference and users' demand, we target to optimize the following long-term objective:

$$\underset{x(t), y(t), c(t): t=0,1,\ldots}{\text{minimize}} \lim_{T \to \infty} \frac{1}{T} \sum_{t \in T} \sum_{v \in \mathcal{V}} E[p_v y_v(t)] \quad (2)$$

subject to

$$x_{iv}(t) \leq y_v(t), \quad \forall\, v \in \mathcal{V}, \forall\, i \in \mathcal{I}, \quad (3)$$

$$\mathbb{1}_{\{c_v(t) = c_u(t)\}} y_v(t) y_u(t) = 0, \quad \forall\, u \in \mathcal{N}_{\mathcal{G}}(v), \forall v \in \mathcal{V}, \quad (4)$$

[1] We define $0/0 = 0$ for the case $\sum_{i \in \mathcal{I}} x_{iv} = 0$.

$$\sum_{v\in\mathcal{V}} x_{iv}(t) = 1, \quad \forall\, i \in \mathcal{I}, \tag{5}$$

$$\lim_{T\to\infty} \frac{1}{T}\sum_{t\in T}\sum_{i\in\mathcal{I}} E[q_i(t)] \le \infty. \tag{6}$$

In the above, control variables $x(t), y(t), c(t)$ are in the following domains:

$$x(t) = [x_{iv}(t) : i \in \mathcal{I}, v \in \mathcal{V}] \in \{0,1\}^{\mathcal{I}\times\mathcal{V}},$$
$$y(t) = [y_v(t) : v \in \mathcal{V}] \in \{0,1\}^{\mathcal{V}}, \quad c(t) = [c_v(t) : v \in \mathcal{V}] \in \mathcal{C}^{\mathcal{V}}.$$

The objective (2) measures the average power consumption of active APs. The constraint (3) denotes AP v is turned on if user i is assigned to v, and the constraint (4) prevents the interference. In addition, the constraint (5) ensures that every user is within the service area and has to be assigned to only one AP. The constraint (6) enforces the queue stability. We note that it is possible that there is no solution satisfying the constraints (3), (4) and (5). To enhance the feasibility, we assume that every maximal independent set of APs in $\mathcal{G}$ can cover every user. In practical scenarios, one can simply place enough APs to make this assumption hold.

To optimize the long-term objective, we formulate the following short-term optimization using the popular Lyapunov approach [11], [20], [21]:

$$\textbf{SMIP:} \quad \underset{x\in\{0,1\}^{\mathcal{I}\times\mathcal{V}}, y\in\{0,1\}^{\mathcal{V}}, c\in\mathcal{C}^{\mathcal{V}}}{\text{maximize}} \quad f_{\texttt{SMIP}}(x,y)$$

subject to

$$\mathbb{1}_{\{c_v=c_u\}} y_v y_u = 0, \quad \forall\, u \in \mathcal{N}_{\mathcal{G}}(v), \forall\, v \in \mathcal{V},$$
$$x_{iv} \le y_v, \quad \forall\, v \in \mathcal{V}, \forall\, i \in \mathcal{I},$$
$$\sum_{v\in\mathcal{V}} x_{iv} = 1, \quad \forall\, i \in \mathcal{I}.$$

In the above,

$$f_{\texttt{SMIP}}(x,y) \;:=\; \sum_{i\in\mathcal{I}}\sum_{v\in\mathcal{V}} q_i r_{iv} \frac{x_{iv}}{\sum_{j\in\mathcal{I}} x_{jv}} - H\sum_{v\in\mathcal{V}} p_v y_v,$$

where $q_i = q_i(t)$ and $H > 0$ is a parameter adjusting priority between stabilizing average queue and minimizing power consumption, i.e., larger H guarantees less power, but might incur larger queues. Computing an optimal solution of **SMIP** in every time leads to a near optimal power consumption with bounded average queue length since it corresponds to minimizing the upper bound of the long-term objective (2) [22]. However, the optimization task is impossible to solve exactly as we prove its NP-hardness in our extended manuscript [22]. Our goal is to develop a polynomial-time approximation algorithm for **SMIP**, which leads to an approximate solution to the long-term objective (2).

III. Main Result

In this section, we provide the main result of this paper: a polynomial-time approximation algorithm for **SMIP**. To this end, we first introduce the notion of *virtual AP*, where each virtual AP corresponds to a pair of a single actual AP and a channel, i.e., $\widehat{\mathcal{V}} = \mathcal{V}\times\mathcal{C}$ is the set of virtual APs. One can naturally consider an interference graph $\widehat{\mathcal{G}} = (\widehat{\mathcal{V}}, \widehat{\mathcal{E}})$ among virtual APs so that a pair of two virtual APs has an edge if they share the same actual AP with different channels or the same channel with different interfering actual APs in the original interference graph $\mathcal{G}$. Using this notion, the optimization problem **SMIP** can be reformulated as follows:

$$\textbf{SMIP:} \quad \underset{x\in\{0,1\}^{\mathcal{I}\times\widehat{\mathcal{V}}}, \mu\in\{0,1\}^{\widehat{\mathcal{V}}}}{\text{maximize}} \quad f_{\texttt{SMIP}}(x,\mu)$$

subject to

$$\mu_{\widehat{u}} + \mu_{\widehat{v}} \le 1, \quad \forall\, \widehat{u} \in \mathcal{N}_{\widehat{\mathcal{G}}}\,(\widehat{v}), \forall\, \widehat{v} \in \widehat{\mathcal{V}}, \tag{7}$$
$$x_{i\widehat{v}} \le \mu_{\widehat{v}}, \quad \forall\, \widehat{v} \in \widehat{\mathcal{V}}, \forall\, i \in \mathcal{I}, \tag{8}$$
$$\sum_{\widehat{v}\in\widehat{\mathcal{V}}} x_{i\widehat{v}} = 1, \quad \forall\, i \in \mathcal{I}. \tag{9}$$

In the above,

$$f_{\texttt{SMIP}}(x,\mu) \;:=\; \sum_{i\in\mathcal{I}}\sum_{\widehat{v}\in\widehat{\mathcal{V}}} w_{i\widehat{v}} \frac{x_{i\widehat{v}}}{\sum_{i\in\mathcal{I}} x_{i\widehat{v}}} - H\sum_{\widehat{v}\in\widehat{\mathcal{V}}} p_{\widehat{v}}\mu_{\widehat{v}}, \tag{10}$$

where $r_{i\widehat{v}} = r_{iv}$, $p_{\widehat{v}} = p_v$ and $w_{i\widehat{v}} = q_i r_{i\widehat{v}}$ for all $\widehat{v} = (v, c)$. We introduce the following short-term linear programming by relaxing integral constraints:

$$\textbf{SLP:} \quad \underset{z\in[0,1]^{\mathcal{I}\times\widehat{\mathcal{V}}}, \mu\in[0,1]^{\widehat{\mathcal{V}}}}{\text{maximize}} \quad \sum_{i\in\mathcal{I}}\sum_{\widehat{v}\in\widehat{\mathcal{V}}} w_{i\widehat{v}} z_{i\widehat{v}} - H\sum_{\widehat{v}\in\widehat{\mathcal{V}}} p_{\widehat{v}}\mu_{\widehat{v}}$$

subject to

$$\mu_{\widehat{u}} + \mu_{\widehat{v}} \le 1, \quad \forall\, \widehat{u} \in \mathcal{N}_{\widehat{\mathcal{G}}}\,(\widehat{v}), \forall\, \widehat{v} \in \widehat{\mathcal{V}}, \tag{11}$$
$$\sum_{i\in\mathcal{I}} z_{i\widehat{v}} \le \mu_{\widehat{v}}, \quad \forall\, \widehat{v} \in \widehat{\mathcal{V}}, \tag{12}$$
$$\frac{1}{\Omega} \le \sum_{\widehat{v}\in\widehat{\mathcal{V}}} z_{i\widehat{v}} \le 1, \quad \forall\, i \in \mathcal{I}, \tag{13}$$
$$\sum_{\widehat{v}: i\in S_{\widehat{v}}} \mu_{\widehat{v}} \ge 1, \quad \forall\, i \in \mathcal{I}, \tag{14}$$

where $S_{\widehat{v}} = S_v$ for all $\widehat{v} = (v, c)$, and $\Omega = \max_{\widehat{v}\in\widehat{\mathcal{V}}} |S_{\widehat{v}}|$. Here, $\frac{x_{i\widehat{v}}}{\sum_{j\in\mathcal{I}} x_{j\widehat{v}}}$ in **SMIP** is replaced by new variable $z_{i\widehat{v}}$ to induce a linear objective function. The constraints (11), (12) and (13) are relaxations of (7), (8) and (9), respectively. In addition, the constraint (14) is from (8) and (9).

Now, we are ready to describe our polynomial-time approximation algorithm called JIT-WLAN (Just-in-time WLAN).

JIT-WLAN 1 Approximation algorithm for **SMIP**

Input: Queue-weighted matrix w=$\{w_{i\widehat{v}} : i \in \mathcal{I},\ \widehat{v} \in \widehat{\mathcal{V}}\}$,
Power p=$\{p_{\widehat{v}} : \widehat{v}\in\widehat{\mathcal{V}}\}$, Interference graph $\widehat{\mathcal{G}}$=$(\widehat{\mathcal{V}}, \widehat{E})$.
Output: Control variables $\bar{x}$, $\bar{y}$, $\bar{c}$.

G.1. Find an optimal solution of **SLP**: (z^*, μ^*)
G.2. Generate candidate solutions by rounding (z^*, μ^*)

$$(\dot{x}, \dot{y}, \dot{c}) \;\leftarrow\; RoundMax(w, p, z^*, \mu^*, \widehat{\mathcal{G}})$$
$$(\ddot{x}, \ddot{y}, \ddot{c}) \;\leftarrow\; RoundMin(w, p, z^*, \mu^*, \widehat{\mathcal{G}})$$

G.3. Finalize the solution as a follow:

$$(\bar{x}, \bar{y}, \bar{c}) = \begin{cases} (\dot{x}, \dot{y}, \dot{c}) & \text{If } f_{\texttt{SMIP}}(\dot{x}, \dot{y}) > f_{\texttt{SMIP}}(\ddot{x}, \ddot{y}) \\ (\ddot{x}, \ddot{y}, \ddot{c}) & \text{otherwise} \end{cases}$$

The proposed algorithm first finds an optimal solution of **SLP**, and generates two candidate solutions of **SMIP** using two greedy rounding schemes, called RoundMax and RoundMin, where detailed explanations are given in Section III-A and III-B, respectively. Finally, it chooses a better one with respect to the objective of **SMIP**. The proposed rounding schemes are motivated by those used for solving the maximum weighted independent set (MWIS) [12] and the minimum set covering (MSC) [13]. Here, our main intuition is that **SMIP** becomes similar to MWIS and MSC if H is small and large, respectively. This allows us to establish constant-factor approximation guarantees that are presented in Section III-C. One can easily check that the overall running time of JIT-WLAN grows polynomially with respect to the size of input instances.

A. Rounding Scheme I: RoundMax

If H is small, the first-term of (10), i.e.,

$$\sum_{i\in\mathcal{I}}\sum_{\widehat{v}\in\widehat{\mathcal{V}}} w_{i\widehat{v}}\frac{x_{i\widehat{v}}}{\sum_{j\in\mathcal{I}} x_{j\widehat{v}}},$$

becomes dominant. The first rounding scheme, called RoundMax, is designed to focus on maximizing the first-term of (10). The intuition behind the proposed scheme is explained in what follows. We first initialize the output, i.e.,

$$\bar{x}\leftarrow\{0\}^{\mathcal{I}\times\mathcal{V}},\ \bar{y}\leftarrow\{0\}^{\mathcal{V}},\ \bar{c}\leftarrow\{0\}^{\mathcal{V}}.$$

Then, we update the optimal solution (z^*,μ^*) of **SLP**:

<u>*A.1*</u> Initialize μ^*, i.e., $\mu^*\leftarrow\{0\}^{\widehat{\mathcal{V}}}$.

<u>*A.2*</u> For every user i, choose a virtual AP $\widehat{v}$ maximizing $w_{i\widehat{v}}z^*_{i\widehat{v}}$ and update (z^*,μ^*) as follows:

$$z^*_{i\widehat{v}}\leftarrow 1,\quad \mu^*_{\widehat{v}}\leftarrow 1,\quad z^*_{i\widehat{u}}\leftarrow 0,\ \forall\ \widehat{u}\in\widehat{\mathcal{V}}\backslash\{\widehat{v}\}.$$

Now, for every virtual AP $\widehat{v}$, we define the following weight on $\widehat{v}$:

$$\pi_{\widehat{v}}=\sum_{i\in\mathcal{I}} w_{i\widehat{v}}\frac{z^*_{i\widehat{v}}}{\sum_{j\in\mathcal{I}} z^*_{j\widehat{v}}}-Hp_{\widehat{v}}\mu^*_{\widehat{v}}. \tag{15}$$

In order to satisfy the constraint (7), we will find the independent set $\widehat{\mathcal{V}}^{\text{on}}$ in $\widehat{\mathcal{G}}$ such that the total weight $\sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\text{on}}}\pi_{\widehat{v}}$ is maximized. Specifically, we initialize the independent set $\widehat{\mathcal{V}}^{\text{on}}\leftarrow\emptyset$ and update it by repeating *B.1-B.5* until $\sum_{\widehat{v}\in\widehat{\mathcal{V}}}\mu^*_{\widehat{v}}=0$:

<u>*B.1*</u> Choose a virtual AP $\widehat{v}\in\widehat{\mathcal{V}}$ which maximizes

$$\frac{\pi_{\widehat{v}}}{|\{\widehat{u}\in\mathcal{N}_{\widehat{\mathcal{G}}}(\widehat{v})\mid\mu^*_{\widehat{u}}=1\}|+1}.$$

<u>*B.2*</u> For the chosen virtual AP $\widehat{v}=(v,c)$, turn on the actual AP v and set the channel of operation as c, i.e., $\bar{y}_v\leftarrow 1,\ \bar{c}_v\leftarrow c$.

<u>*B.3*</u> For every user i such that $z^*_{i\widehat{v}}>0$, assign user i to actual AP v, i.e., $\bar{x}_{iv}\leftarrow 1$.

<u>*B.4*</u> Add $\widehat{v}$ to $\widehat{\mathcal{V}}^{\text{on}}$ and remove its neighbors from $\widehat{\mathcal{V}}$, i.e.,

$$\widehat{\mathcal{V}}^{\text{on}}\leftarrow\widehat{\mathcal{V}}^{\text{on}}\cup\{\widehat{v}\},\ \widehat{\mathcal{V}}\leftarrow\widehat{\mathcal{V}}\backslash(\mathcal{N}_{\widehat{\mathcal{G}}}(\widehat{v})\cup\{\widehat{v}\}).$$

<u>*B.5*</u> Update the interference graph $\widehat{\mathcal{G}}$ by the subgraph induced by the modified set of virtual APs $\widehat{\mathcal{V}}$.

We design the above procedure inspired by the greedy algorithm [12] for computing an approximation solution of MWIS, and the current solution $(\bar{x},\bar{y},\bar{c})$ satisfies the constraint (7). However, it might not be feasible for **SMIP** due to the constraint (9), i.e., some users are not assigned. To enforce feasibility, we assign remaining users by repeating *C.1-C.4* until all users are assigned:

<u>*C.1*</u> Choose a user i randomly from the set of unassigned users, i.e., $\{i\in\mathcal{I}\mid\sum_{v\in\mathcal{V}}\bar{x}_{iv}=0\}$.

<u>*C.2*</u> For every virtual AP $\widehat{v}=(v,c)\in\widehat{\mathcal{V}}^{\text{on}}\cup\widehat{\mathcal{V}}$ such that $w_{i\widehat{v}}>0$, estimate the marginal gain ψ of assigning user i to AP v.

$$\widetilde{x}\leftarrow\bar{x},\ \widetilde{y}\leftarrow\bar{y},\ \widetilde{x}_{iv}\leftarrow 1,\ \widetilde{y}_v\leftarrow 1,$$
$$\psi(\widehat{v})\leftarrow f_{\texttt{SMIP}}(\widetilde{x},\widetilde{y})-f_{\texttt{SMIP}}(\bar{x},\bar{y}).$$

<u>*C.3*</u> Choose a virtual AP $\widehat{v}=(v,c)$ maximizing $\psi(\widehat{v})$, and assign user i to actual AP v, i.e., $\bar{x}_{iv}\leftarrow 1$.

<u>*C.4*</u> If $\widehat{v}=(v,c)\notin\widehat{\mathcal{V}}^{\text{on}}$, update $\bar{y}_v,\bar{c}_v,\widehat{\mathcal{V}},\widehat{\mathcal{V}}^{\text{on}}$ following *B.2* and *B.4*.

Finally, one can check that $(\bar{x},\bar{y},\bar{c})$ satisfies all constraints of **SMIP**.

B. Rounding Scheme II: RoundMin

In contrast with RoundMax, the second rounding scheme is designed to focus on minimizing the second-term of (10), i.e.,

$$\sum_{\widehat{v}\in\widehat{\mathcal{V}}} p_{\widehat{v}}\mu_{\widehat{v}}.$$

The intuition behind the proposed scheme is explained in what follows. We first initialize the output, i.e.,

$$\bar{x}\leftarrow\{0\}^{\mathcal{I}\times\mathcal{V}},\ \bar{y}\leftarrow\{0\}^{\mathcal{V}},\ \bar{c}\leftarrow\{0\}^{\mathcal{V}}.$$

Then, using the optimal solution (z^*,μ^*) of **SLP**, we will find the independent set $\widehat{\mathcal{V}}^{\text{on}}$ in $\widehat{\mathcal{G}}$ which covers every user minimizing power consumption as follows:

<u>*D.1*</u> Initialize the independent set $\widehat{\mathcal{V}}^{\text{on}}$ by selecting all virtual APs $\widehat{v}$ with $\mu^*_{\widehat{v}}>\frac{1}{2}$, i.e., $\widehat{\mathcal{V}}^{\text{on}}\leftarrow\{\widehat{v}\in\widehat{\mathcal{V}}\mid\mu^*_{\widehat{v}}>1/2\}$.

<u>*D.2*</u> For every virtual AP $\widehat{v}\in\widehat{\mathcal{V}}^{\text{on}}$, remove its neighbors from $\widehat{\mathcal{V}}$, i.e., $\widehat{\mathcal{V}}\leftarrow\widehat{\mathcal{V}}\backslash(\mathcal{N}_{\widehat{\mathcal{G}}}(\widehat{v})\cup\{\widehat{v}\})$.

<u>*D.3*</u> Define the set of uncovered users $\mathcal{I}^{\text{un}}$ by removing users covered by virtual AP in $\widehat{\mathcal{V}}^{\text{on}}$ from $\mathcal{I}$, i.e., $\mathcal{I}^{\text{un}}\leftarrow\mathcal{I}\setminus(\cup_{\widehat{\mathcal{V}}^{\text{on}}}S_{\widehat{v}})$.

<u>*D.4*</u> Choose a virtual AP $\widehat{v}\in\widehat{\mathcal{V}}$ which minimizes

$$\frac{p_{\widehat{v}}}{|S_{\widehat{v}}\cap\mathcal{I}^{\text{un}}|}.$$

<u>*D.5*</u> Add $\widehat{v}$ to $\widehat{\mathcal{V}}^{\text{on}}$, i.e., $\widehat{\mathcal{V}}^{\text{on}}\leftarrow\widehat{\mathcal{V}}^{\text{on}}\cup\{\widehat{v}\}$. Then, remove its neighbors and users covered by it from $\widehat{\mathcal{V}}$ and $\mathcal{I}^{\text{un}}$, respectively, i.e.,

$$\widehat{\mathcal{V}}\leftarrow\widehat{\mathcal{V}}\backslash(\mathcal{N}_{\widehat{\mathcal{G}}}(\widehat{v})\cup\{\widehat{v}\}),\ \mathcal{I}^{\text{un}}\leftarrow\mathcal{I}^{\text{un}}\backslash S_{\widehat{v}}.$$

<u>*D.6*</u> Repeat *D.4-D.5* until all users are covered, i.e., $\mathcal{I}^{\text{un}}=\emptyset$.

We design the above procedure inspired by the greedy algorithm [13] for computing an approximation solution of MSC. Next, we decide the user association as follows:

<u>*E.1*</u> For every virtual AP $\widehat{v}\notin\widehat{\mathcal{V}}^{\text{on}}$, replace its queue-weighted achievable rate by zero, i.e., $w_{i\widehat{v}}\leftarrow 0,\quad\forall\ i\in\mathcal{I}$.

<u>*E.2*</u> Find a new optimal solution (z^*,μ^*) of **SLP** using the updated w.

<u>*E.3*</u> For every user i, choose a virtual AP $\widehat{v} = (v, c)$ maximizing $w_{i\widehat{v}} z^*_{i\widehat{v}}$ and update $\bar{x}_{iv} \leftarrow 1,\ \bar{y}_v \leftarrow 1,\ \bar{c}_v \leftarrow c$.

Finally, one can check that the current solution $(\bar{x}, \bar{y}, \bar{c})$ satisfies all constraints of **SMIP**.

C. Performance Guarantees

We show that JIT-WLAN has the following constant approximation ratios for **SMIP** depending on parameter regimes, where we let $(\bar{x}, \bar{y})$ be the output of JIT-WLAN and (x^*, y^*) be the optimal solution of **SMIP**.

Theorem 3.1: If $\alpha = \frac{w_{\min}}{Hp_{\max}} > 1$, then $f_{\texttt{SMIP}}(x^*, y^*) > 0$ and

$$f_{\texttt{SMIP}}(x^*, y^*) \ = \ O\left(\frac{\alpha}{\alpha - 1}\right) \cdot f_{\texttt{SMIP}}(\bar{x}, \bar{y}),$$

where $p_{\max} = \max_{v \in \mathcal{V}} p_v$ and $w_{\min} = \min_{i \in \mathcal{I}, v \in \mathcal{V}} \{w_{iv} \mid w_{iv} > 0\ \}$.

Theorem 3.2: If $\beta = \frac{Hp_{\min}}{w_{\max}} > 1$, then $f_{\texttt{SMIP}}(x^*, y^*) < 0$ and

$$O\left(\frac{\beta}{\beta - 1}\right) \cdot f_{\texttt{SMIP}}(x^*, y^*) \ = \ f_{\texttt{SMIP}}(\bar{x}, \bar{y}),$$

where $p_{\min} = \min_{v \in \mathcal{V}} p_v$ and $w_{\max} = \max_{i \in \mathcal{I}, v \in \mathcal{V}} w_{iv}$.

The proofs of the above theorems are given in Section IV-A and Section IV-B, respectively. We note that the objective $f_{\texttt{SMIP}}$ is positive, i.e., Theorem 3.1, and negative, i.e., Theorem 3.2, for small and large H compared to weight $[w_{i\widehat{v}} = q_i r_{i\widehat{v}}]$, respectively. Hence, the conditions $\frac{w_{\min}}{Hp_{\max}} > 1, \frac{Hp_{\min}}{w_{\max}} > 1$ are inevitable to obtain such approximation ratios since $f_{\texttt{SMIP}}$ can be zero without them. Nevertheless, such conditions are merely added due to the technical reason and the proposed algorithm is designed to work well for any value of H. We demonstrate its performance via extensive simulations in Section V, in particular focusing on how well it optimizes the long-term objective (2).

IV. Proofs of Theorems

A. Proof of Theorem 3.1

We first present the proof of Theorem 3.1 by analyzing the approximation ratio of RoundMax. One can easily check that if $\alpha = \frac{w_{\min}}{Hp_{\max}} > 1$, then the following inequality is hold:

$$\sum_{i \in \mathcal{I}} w_{iv} \frac{x_{iv}}{\sum_{j \in \mathcal{I}} x_{jv}} \geq w_{\min} > Hp_{\max} \geq Hp_v y_v, \tag{16}$$

for every $v \in \mathcal{V}$ such that $y_v = 1$. This inequality (16) implies if $\alpha = \frac{w_{\min}}{Hp_{\max}} > 1$, then the objective $f_{\texttt{SMIP}}$ is always positive. Let $(\bar{x}, \bar{y})$ be the output of this rounding scheme and $\mathcal{V}^{\text{on}}$ be the set of actual APs which are chosen by repeating *B.1-B.5*.

For every AP $v \in \mathcal{V}^{\text{on}}$, we denote $\mathcal{I}^{\text{B}}_v$ and $\mathcal{I}^{\text{C}}_v$ a set of users who are assigned to AP v at *B.3* and *C.3*, respectively. From (16), we have

$$\begin{aligned}
&\sum_{i \in \mathcal{I}} w_{iv} \frac{\bar{x}_{iv}}{\sum_{j \in \mathcal{I}} \bar{x}_{jv}} - Hp_v \bar{y}_v \\
&= \sum_{i \in \mathcal{I}^{\text{B}}_v} \frac{w_{iv}}{|\mathcal{I}^{\text{B}}_v| + |\mathcal{I}^{\text{C}}_v|} + \sum_{i \in \mathcal{I}^{\text{C}}_v} \frac{w_{iv}}{|\mathcal{I}^{\text{B}}_v| + |\mathcal{I}^{\text{C}}_v|} - Hp_v \bar{y}_v \\
&= \frac{|\mathcal{I}^{\text{B}}_v|}{|\mathcal{I}^{\text{B}}_v| + |\mathcal{I}^{\text{C}}_v|} \left(\sum_{i \in \mathcal{I}^{\text{B}}_v} \frac{w_{iv}}{|\mathcal{I}^{\text{B}}_v|} - Hp_v \bar{y}_v \right) \\
&\quad + \frac{|\mathcal{I}^{\text{C}}_v|}{|\mathcal{I}^{\text{B}}_v| + |\mathcal{I}^{\text{C}}_v|} \left(\sum_{i \in \mathcal{I}^{\text{C}}_v} \frac{w_{iv}}{|\mathcal{I}^{\text{C}}_v|} - Hp_v \bar{y}_v \right) \\
&\geq \frac{|\mathcal{I}^{\text{B}}_v|}{|\mathcal{I}^{\text{B}}_v| + |\mathcal{I}^{\text{C}}_v|} \left(\sum_{i \in \mathcal{I}^{\text{B}}_v} \frac{w_{iv}}{|\mathcal{I}^{\text{B}}_v|} - Hp_v \bar{y}_v \right),
\end{aligned}$$

for every $v \in \mathcal{V}^{\text{on}}$. Using above inequality, one can write

$$\begin{aligned}
f_{\texttt{SMIP}}(\bar{x}, \bar{y}) &= \sum_{v \in \mathcal{V}} \left(\sum_{i \in \mathcal{I}} w_{iv} \frac{\bar{x}_{iv}}{\sum_{j \in \mathcal{I}} \bar{x}_{jv}} - Hp_v \bar{y}_v \right) \\
&\geq \sum_{v \in \mathcal{V}^{\text{on}}} \left(\sum_{i \in \mathcal{I}} w_{iv} \frac{\bar{x}_{iv}}{\sum_{j \in \mathcal{I}} \bar{x}_{jv}} - Hp_v \bar{y}_v \right) \\
&\geq \sum_{v \in \mathcal{V}^{\text{on}}} \frac{|\mathcal{I}^{\text{B}}_v|}{|\mathcal{I}^{\text{B}}_v| + |\mathcal{I}^{\text{C}}_v|} \left(\sum_{i \in \mathcal{I}^{\text{B}}_v} \frac{w_{iv}}{|\mathcal{I}^{\text{B}}_v|} - Hp_v \bar{y}_v \right) \\
&\geq \frac{1}{\Omega} \sum_{v \in \mathcal{V}^{\text{on}}} \left(\sum_{i \in \mathcal{I}^{\text{B}}_v} \frac{w_{iv}}{|\mathcal{I}^{\text{B}}_v|} - Hp_v \bar{y}_v \right) = \frac{1}{\Omega} \sum_{\widehat{v} \in \widehat{\mathcal{V}}^{\text{on}}} \pi_{\widehat{v}},
\end{aligned}$$

where $\widehat{\mathcal{V}}^{\text{on}}$ is a set of virtual APs which are chosen by repeating *B.1-B.5* and we recall that r_{iv} is the achievable rate from AP v to user i, $\Omega = \max_{v \in \mathcal{V}} |\{i \in \mathcal{I} | r_{iv} > 0\}|$ and $\pi_{\widehat{v}}$ is as defined in (15). One can check that the above inequality is hold by the definition of $\pi_{\widehat{v}}$ with the fact that $\Omega \geq |\mathcal{I}^{\text{B}}_v| + |\mathcal{I}^{\text{C}}_v|$.

Now, we state the following key lemma, where its proof is presented in our extended manuscript [22].

Lemma 4.1: Let $\widehat{\mathcal{V}}^{\text{on}}$ be a set of virtual APs which are chosen by repeating *B.1-B.5*, and (x^*, y^*) be an optimal solution of **SMIP**. If $\alpha = \frac{w_{\min}}{Hp_{\max}} > 1$, then the following inequality is hold:

$$\frac{\alpha - 1}{\alpha \Lambda \Omega |\mathcal{C}| (\Delta + |\mathcal{C}| + 1)} f_{\texttt{SMIP}}(x^*, y^*) \leq \sum_{\widehat{v} \in \widehat{\mathcal{V}}^{\text{on}}} \pi_{\widehat{v}},$$

where we recall that $|\mathcal{C}|$ is the number of non-overlapping channels, $\Lambda = \max_{i \in \mathcal{I}} |\{v \in \mathcal{V} | r_{iv} > 0\}|$ and $\Delta = \max_{v \in \mathcal{V}} |\mathcal{N}_{\mathcal{G}}(v)|$.

From Lemma 4.1, it follows that

$$\frac{f_{\texttt{SMIP}}(x^*, y^*)}{f_{\texttt{SMIP}}(\bar{x}, \bar{y})} \leq \frac{\alpha \Lambda |\mathcal{C}| (\Delta + |\mathcal{C}| + 1) \Omega^2}{\alpha - 1}.$$

As already mentioned, in many practical scenarios, $|\mathcal{C}|$, Ω, Λ and $\Delta = O(1)$. Therefore, we can conclude

$$f_{\texttt{SMIP}}(x^*, y^*) \ = \ O\left(\frac{\alpha}{\alpha - 1}\right) \cdot f_{\texttt{SMIP}}(\bar{x}, \bar{y}).$$

This completes the proof of Theorem 3.1.

B. Proof of Theorem 3.2

In this section, we present the proof of Theorem 3.2 by analyzing the approximation ratio of RoundMin. Using the optimal solution (z^*, μ^*) of **SLP**, we let $\widehat{\mathcal{V}}^{\text{LP}} = \{\widehat{v} \in \widehat{\mathcal{V}} | \mu^*_{\widehat{v}} > 1/2\}$ be a set of virtual APs chosen at *D.1* and $\widehat{\mathcal{V}}^{\text{G}} = \widehat{\mathcal{V}}^{\text{on}} \backslash \widehat{\mathcal{V}}^{\text{LP}}$ be a set of virtual APs chosen by a greedy approach as described in *D.4-D.6*. Then, the sum of power consumption has the following upper bound:

$$\sum_{v \in \mathcal{V}} p_v \bar{y}_v \leq \sum_{\widehat{v} \in \widehat{\mathcal{V}}^{\text{LP}}} p_{\widehat{v}} + \sum_{\widehat{v} \in \widehat{\mathcal{V}}^{\text{G}}} p_{\widehat{v}},$$

where $(\bar{x}, \bar{y})$ be the output of RoundMin. First, we first show the upper bound of $\sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\texttt{LP}}} p_{\widehat{v}}$, as stated in the next lemma whose proof is presented in our extended manuscript [22].

Lemma 4.2: Let (x^*, y^*) be an optimal solution of **SMIP**. If $\beta = \frac{Hp_{\min}}{w_{\max}} > 1$, then a following inequality is hold:

$$\sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\texttt{LP}}} p_{\widehat{v}} < \frac{2\beta}{H(1-\beta)} f_{\texttt{SMIP}}(x^*, y^*).$$

Next, we will find the upper bound of $\sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\texttt{G}}} p_{\widehat{v}}$, Let $\mathcal{I}^{\text{un}} = \mathcal{I}\backslash(\cup_{\widehat{\mathcal{V}}^{\texttt{LP}}} S_{\widehat{v}})$ and $\mathcal{I}^{\text{un}}_{\widehat{v}}$ be a set of users that are uncovered by greedy approach when a virtual AP $\widehat{v} \in \widehat{\mathcal{V}}^{\texttt{G}}$ is picked. Then, we define the cost of covering user i:

$$\zeta_i = \begin{cases} \frac{p_{\widehat{v}}}{|S_{\widehat{v}}\cap\mathcal{I}^{\text{un}}_{\widehat{v}}|} & \text{If user } i \in \mathcal{I}^{\text{un}} \text{ is first covered by } \widehat{v} \in \widehat{\mathcal{V}}^{\texttt{G}} \\ 0 & \text{otherwise.} \end{cases}$$

Then, one can write the sum of power consumption of virtual APs using ζ as follows:

$$\sum_{i\in\mathcal{I}} \zeta_i = \sum_{i\in\mathcal{I}^{\text{un}}} \zeta_i = \sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\texttt{G}}} \sum_{i\in S_{\widehat{v}}\cap\mathcal{I}^{\text{un}}_{\widehat{v}}} \frac{p_{\widehat{v}}}{|S_{\widehat{v}}\cap\mathcal{I}^{\text{un}}_{\widehat{v}}|} = \sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\texttt{G}}} p_{\widehat{v}}. \quad (17)$$

Now, we state the following lemma on the upper bound of ζ, where its proof is presented in our extended manuscript [22].

Lemma 4.3: For every virtual AP $\widehat{v} \in \widehat{\mathcal{V}}$, the cost ζ has the following upper bound:

$$\sum_{i\in S_{\widehat{v}}} \zeta_i \le \frac{|S_{\widehat{v}}\cap\mathcal{I}^{\text{un}}| p_{\max}}{p_{\min}} p_{\widehat{v}}$$

where we recall that $p_{\min} = \min_{v\in\mathcal{V}} p_v$ and $p_{\max} = \max_{v\in\mathcal{V}} p_v$.

Let $\widehat{\mathcal{V}}^*$ be a set of virtual APs which indicate the optimal solution of **SMIP**, i.e., $\widehat{\mathcal{V}}^* = \{\widehat{v} = (v, c) \in \widehat{\mathcal{V}} \mid y^*_v = 1,\ c^*_v = c\}$. From Lemma 4.3 and (17), we have

$$\begin{aligned} \sum_{v\in\mathcal{V}} p_v y^*_v &= \sum_{\widehat{v}\in\widehat{\mathcal{V}}^*} p_{\widehat{v}} \ge \sum_{\widehat{v}\in\widehat{\mathcal{V}}^*} \frac{p_{\min}}{|S_{\widehat{v}}\cap\mathcal{I}^{\text{un}}| p_{\max}} \sum_{i\in S_{\widehat{v}}} \zeta_i \\ &\ge \frac{p_{\min}}{\Omega^{\text{un}} p_{\max}} \sum_{\widehat{v}\in\widehat{\mathcal{V}}^*} \sum_{i\in S_{\widehat{v}}} \zeta_i \ge \frac{p_{\min}}{\Omega^{\text{un}} p_{\max}} \sum_{i\in\mathcal{I}} \zeta_i \\ &= \frac{p_{\min}}{\Omega^{\text{un}} p_{\max}} \sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\texttt{G}}} p_{\widehat{v}}, \end{aligned} \quad (18)$$

where $\Omega^{\text{un}} = \max_{\widehat{v}\in\widehat{\mathcal{V}}} |S_{\widehat{v}}\cap\mathcal{I}^{\text{un}}|$.

One can easily check that if $\beta = \frac{Hp_{\min}}{w_{\max}} > 1$, then the following inequality is hold:

$$\sum_{i\in\mathcal{I}} w_{iv} \frac{x^*_{iv}}{\sum_{j\in\mathcal{I}} x^*_{jv}} \le w_{\max} = \frac{Hp_{\min}}{\beta} \le \frac{Hp_v y^*_v}{\beta}, \quad (19)$$

for every $v \in \mathcal{V}$ such that $y^*_v = 1$.

From (19), we extend (18) as follows

$$\begin{aligned} f_{\texttt{SMIP}}(x^*, y^*) &\le \sum_{v\in\mathcal{V}} \left(\frac{Hp_v}{\beta} - Hp_v\right) y^*_v \\ &\le -\frac{\beta-1}{\beta} H \sum_{v\in\mathcal{V}} p_v y^*_v \\ &\le -\frac{\beta-1}{\beta} \frac{p_{\min}}{\Omega^{\text{un}} p_{\max}} H \sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\texttt{G}}} p_{\widehat{v}}. \end{aligned} \quad (20)$$

Combining the inequality (20) with Lemma 4.2 leads to the following inequality:

$$\begin{aligned} f_{\texttt{SMIP}}(\bar{x}, \bar{y}) &\ge -H \sum_{v\in\mathcal{V}} p_v \bar{y}_v \ge -H \left(\sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\texttt{LP}}} p_{\widehat{v}} + \sum_{\widehat{v}\in\widehat{\mathcal{V}}^{\texttt{G}}} p_{\widehat{v}} \right) \\ &\ge \frac{2\beta}{\beta-1} f_{\texttt{SMIP}}(x^*, y^*) + \frac{\beta}{\beta-1} \frac{\Omega^{\text{un}} p_{\max}}{p_{\min}} f_{\texttt{SMIP}}(x^*, y^*) \\ &\ge \left(2 + \frac{\Omega^{\text{un}} p_{\max}}{p_{\min}}\right) \frac{\beta}{\beta-1} f_{\texttt{SMIP}}(x^*, y^*). \end{aligned}$$

Since $\frac{p_{\max}}{p_{\min}}$ and Ω^{un} do not scale with the number of APs, it follows that

$$O\left(\frac{\beta}{\beta-1}\right) \cdot f_{\texttt{SMIP}}(x^*, y^*) \ = \ f_{\texttt{SMIP}}(\bar{x}, \bar{y}).$$

This completes the proof of Theorem 3.2.

V. Simulation Results

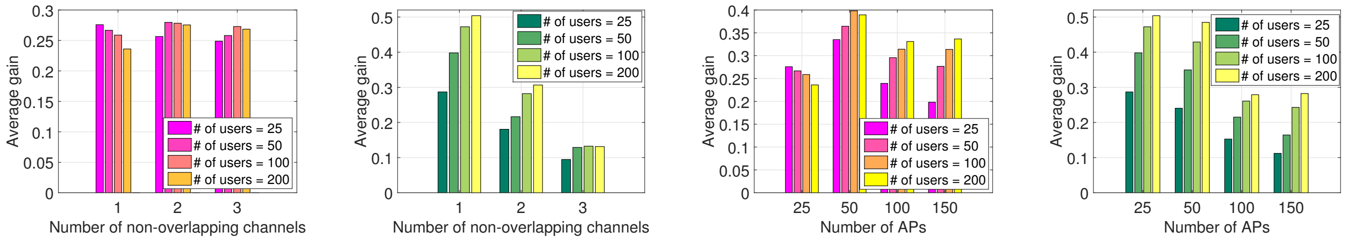
To verify the performance of the proposed algorithm, we use the following algorithms for comparison.

1) **Traffic Aware (Traf-Aw).** This runs I iterations where at each iteration a (user, AP) pair which maximizes $f_{\texttt{SMIP}}$ is selected, and it removes neighbors of selected AP from the graph to avoid the interference.
2) **Power Minimization (Pow-Min).** This selects an AP using a greedy method developed in [13], assigns users to it, and removes its neighbors from the graph.
3) **Throughput Maximization (Thr-Max).** This is similar to Traf-Aw but selects a (user, AP) pair which only maximizes the first-term of $f_{\texttt{SMIP}}$ to maximize the throughput.

A. Short-term Performance

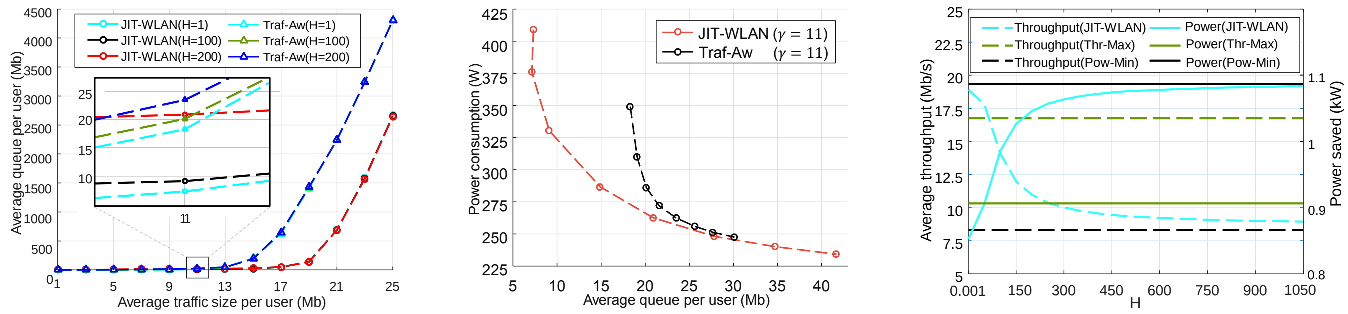
We first verify how well JIT-WLAN solves **SMIP** by comparing the average gain of our algorithm against Traf-Aw. Specifically, let $(\bar{x}, \bar{y})$ and $(\widehat{x}, \widehat{y})$ be the output of JIT-WLAN and Traf-Aw, respectively. To quantify the average gain, we use $\frac{f_{\texttt{SMIP}}(\bar{x},\bar{y}) - f_{\texttt{SMIP}}(\widehat{x},\widehat{y})}{f_{\texttt{SMIP}}(\widehat{x},\widehat{y})}$ and $\frac{f_{\texttt{SMIP}}(\widehat{x},\widehat{y}) - f_{\texttt{SMIP}}(\bar{x},\bar{y})}{f_{\texttt{SMIP}}(\widehat{x},\widehat{y})}$ when H is small and large, respectively. In order to strictly validate the performance, we use a topology data set of Google WiFi network [23]. Based on WiFi data sheets in [7], [24], the baseline power consumption is randomly selected from 5-12W. We run 300 iterations where at each iteration we place users randomly according to the uniform distribution and select the queue-weighted achievable rate w_{iv} from 0-100 according to a location of users.

From Fig.1(a), we can note that the number of non-overlapping channels has relatively small effect on the performance when H is large. The average gain fluctuates between 0.23 and 0.27 since the number of non-overlapping channel only changes the number of active APs. Also, the saturation of performance over the number of users is observed. However, Fig.1(b) shows that the average gain drops from 0.5 to 0.1 as the number of non-overlapping channels increases when H is small. We also note that the number of APs has relatively large effect on the performance as shown in Fig.1(c) and 1(d). Although our algorithm always better than Traf-Aw, the average gain is slightly reduced as the number of APs increases.



(a) Average gain for varying number of channels (# of APs=25, H=10^8)
(b) Average gain for varying number of channels (# of APs=25, H=10^{-8})
(c) Average gain for varying number of APs (# of channels=1, H=10^8)
(d) Average gain for varying number of APs (# of channels=1, H=10^{-8})

Fig. 1. Short-term performance evaluation of JIT-WLAN over the Google WiFi network topology.



(a) Average queue length comparison by varying the average file size $\gamma = 1 \sim 25$Mb
(b) Power-queue tradeoff by varying the power-queue tradeoff parameter $H = 0.1 \sim 400$
(c) Throughput and saved power comparison for varying H ($\gamma = 9$Mb)

Fig. 2. Long-term performance evaluation of JIT-WLAN over the Google WiFi network topology composed of 100 users and 150 APs.

B. Long-term Performance

Next, we verify that JIT-WLAN can achieve the long term objective. We assume that one non-overlapping channel is available, and consider a network topology composed of 100 users and 150 APs which consume 5-12W. Each user walks in a random direction inside a service area of WiFi networks and the achievable rate varies from 18 to 54Mb/s according to the location. For traffic generation, we assume that traffic arrivals follow a Poisson process with arrival rate 1 and each arrival file follows exponential distribution with average size γ. Changing the value of γ, we simulate our algorithm with different H for 1 thousand timeslots against other tested algorithms.

We first compare an average queue length as shown in Fig.2(a). As expected, our algorithm can stabilize the queue better than Traf-Aw since it can achieve high throughput. In Fig.2(b), we plot the power-queue tradeoff curves by varying the value of H. One can note that Traf-Aw consumes more power than our algorithm to have a same queue length. We also compare the average throughput and saved power with Pow-Min and Thr-Max by varying H. Fig.2(c) shows that our algorithm not only saves more power as much as Pow-Min, but also achieves higher throughput than Thr-Max with appropriate choices of H. It indicates that our algorithm outperforms other tested algorithms despite taking both power saving and throughput maximization into account.

VI. Implementation

We have built a prototype of a centralized management system to verify how our algorithm manages WiFi networks effectively and efficiently in a real-world environment. We leverage SDN framework to implement the proposed algorithm and control policies as well as seamless handover. As illustrated in Fig.3(a), a prototype architecture consists of three main components: JIT Manager, SDN-enabled APs, and user devices.

A. JIT Manager

JIT Manager is an SDN control platform that runs JIT-WLAN and manages user associations and a power of each AP as a result of the algorithm. We implement the JIT Manager on Floodlight version 0.9 [25], which is one of the popular SDN controllers written in Java. There are three modules in the controller: Status Receiver, JIT Optimizer, and Odin Master.

Status Receiver. A main role of this module is to collect achievable data rates from user devices. The module periodically sends a scanning request to each user device and receives scanning results (i.e., SSIDs, RSSIs). Then, it makes an achievable rate matrix using a mapping table that converts an RSSI to an achievable rate [4], and sends the matrix to JIT Optimizer to run the algorithm.

JIT Optimizer. We implement this module based on CPLEX, JAVA based linear programming solver. For each period, JIT Optimizer sends OpenFlow flow statistics requests to connected APs to collect the number of incoming and outgoing packets, and receives the achievable matrix from Status Receiver. Then, this module runs the proposed algorithm to draw user-AP associations, and those results will be sent to Odin Master for seamless handovers. After completing user associations, JIT Optimizer sends control messages (i.e., channel control and operation mode control) to each AP.

Odin Master. To conduct seamless handover operations, we leverage Odin framework developed in [16]. In Odin framework, cooperating with Odin Agent, Odin Master decouples a user-AP association and authentication from a physical connection using a concept of virtual AP. Therefore, we can perform a seamless handover without requesting a re-association and a re-authentication.

B. SDN-enabled AP and User Device

SDN-enabled AP. In an SDN architecture, an SDN controller manages SDN switches through OpenFlow protocol [26]. However, unfortunately, since most of commercial APs do not support OpenFlow, we installed Open vSwitch (OVS) version 1.9.0 on OpenWrt firmware version 12.09 to handle OpenFlow messages from the controller in APs. The SDN-enabled AP has two main agents (i.e., Odin Agent and JIT Agent) to perform specific functions. The Odin Agent is one of the components of Odin framework. It creates and assigns a light virtual AP (LVAP) to a user to support seamless handover by moving the LVAP to another SDN-enabled AP. JIT Agent is implemented as a shell script, which receives a control message from the controller and changes the operating mode of the AP.

User devices. Since WiFi network scanning results from each user are required to infer achievable data rates, we develop an application for a user device to collect the scanning results. It periodically scans WiFi networks and sends scanning results to Status Receiver in the controller.

C. Issues in Implementing a Testbed

In this section, we discuss issues that we faced while implementing our prototype and how we resolve them.

Queue calculation. It is hard to know exact queue lengths of every user device since JIT Manager can only count incoming packets of each flow and each switch total outgoing packet using SDN. Instead, we develop a simple and natural algorithm (see our extended manuscript [22]) that conjectures queue lengths from available data.

AP operation control. In our framework, we control an operating mode and a channel of APs. We expect that a power consumption of APs is dramatically reduced by using Wake-on-LAN (WoL), which turns on/off the AP through a network message. However, we implement a shell script to turn on or off a WiFi interface instead of the WoL since our AP model does not support the WoL completely. For a channel control, we note that a *iwconfig* command can be used to set an operating channel of the AP. However, when we change the operating channel using *iwconfig*, an inevitable performance degradation occurs from re-association. In this reason, we omit to implement the channel selection from this paper.

VII. Evaluation

A. Experiment Setup

We deploy a wireless testbed as shown in Fig.3(b). It consists of one controller machine, three Buffalo-WZR-HP-AG300H APs, two Galaxy S3 GT-I9305 android smartphones and one data server. AP 2 interferes with AP 1 and 3 since it has the overlapped interference region with them. As performance metrics, we measure UDP throughput from received data generated by the data server and power consumption using a suitable power meter called PowerManager B310-W2 [27].

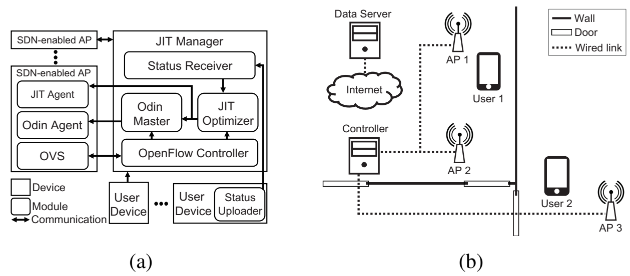


Fig. 3. The architecture of the prototype management system (a) and topology of the testbed (b).

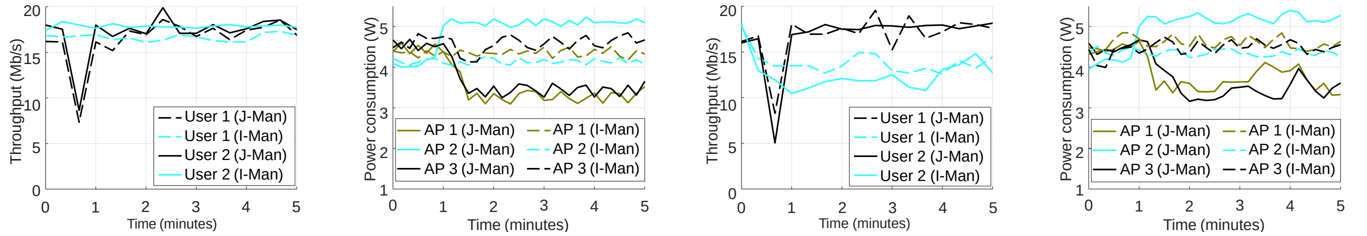
We test the following management systems for evaluation.

1) **JIT Management (J-Man).** This is the management system described in Section VI. In J-Man, the JIT Manager operates with five minutes-cycle as following processes: In the first minute, it turns on all APs and collects WiFi scanning results from users. Using collected data, it runs JIT-WLAN for saving energy and controls AP operation mode and user-AP association according to results of the algorithm. After this first minute, it sleeps for four minutes.
2) **Idle Management (I-Man).** In this management system, JIT Manager operates in idle mode. All APs are always on, and users maintain a current association.

B. Experiment Results

Stationary user scenario. We first evaluate our management system when users are at a fixed location and do not move on. User 1 and 2 are placed as shown in Fig.3(b), and associated with AP 1 and 3, respectively. We generate constant bit rate UDP flow of 20Mb/s to each user and manage the network using J-Man. We repeat the above procedure using I-Man and compare the performance in terms of throughput and power consumption. When the network is managed by J-Man, JIT Manager dynamically hands-off User 1 and 2 to AP 2 and turns off AP 1 and 3 in order to save power consumption. Although the throughput of both users slightly drops due to the re-association shown in Fig.4(a), Fig.4(b) shows J-Man achieves 8% aggregate power saving. Since we only turn off a WiFi interface (see Section VI-B) and the scale of our test bed is small, the amount of energy saved in J-Man is relatively small. However, the experimental results indicate that J-Man can reduce the wastage of energy maintaining a high throughput, and in the case of operating a large number of APs, the saving effect will be much higher than this test case.

Mobile user scenario. We also evaluate our management system when users move on. The experimental setup is the same as the previous scenario except that User 1 and 2 move into AP 2 in the first minute. In this scenario, J-Man manages the network in the same way as the previous scenario. As a result, the throughput in J-Man varies from 15 to 20Mb/s except the handoff duration, whereas the throughput in I-Man drops to 10-13Mb/s as both users move into AP 2 shown in Fig.4(c). In addition, as shown in Fig.4(d), J-Man effectively



(a) Throughput received by users in stationary user scenario.

(b) Power consumptions of APs in stationary user scenario.

(c) Throughput received by users in mobile user scenario.

(d) Power consumptions of APs in mobile user scenario.

Fig. 4. Time series plot of evaluation results in each scenarios.

saves power by turning off useless APs. These results imply that the proposed management system can achieve both high throughput and energy saving in practice.

VIII. Conclusion

Dense deployment of WLAN APs causes two critical issues: wastage of energy and severe interference. In this paper, we design and analyze a constant-factor polynomial-time approximation algorithm which jointly optimizes power-operation modes, channel selections and user-AP associations for improving energy efficiency and avoiding interference without sacrificing users' demands. We believe that the proposed algorithm will provide promising solution to manage WLAN infrastructures more effectively.

IX. Acknowledgement

This work makes use of results produced by the SmartFIRE project, which is supported by the International Research and Development Program of the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (MSIP, Korea) (Grant number: K2013078191) and the Seventh Framework Programme (FP7) funded by the European Commission (Grant number: 611165), and the Institute for Information and Communications Technology Promotion (IITP) funded by the Korea government (MSIP) (Grant B0190-15-2017, Resilient/Fault-Tolerant Autonomic Networking Based on Physicality, Relationship and Service Semantic of IoT Devices).

References

[1] A. P. Jardosh, K. Papagiannaki, E. M. Belding, K. C. Almeroth, G. Iannaccone, and B. Vinnakota, "Green WLANs: on-demand WLAN infrastructures," *Mobile Networks and Applications*, vol. 14, no. 6, pp. 798–814, 2009.

[2] A. Akella, G. Judd, S. Seshan, and P. Steenkiste, "Self-management in chaotic wireless deployments," *Wireless Networks*, vol. 13, no. 6, pp. 737–755, 2007.

[3] I. Broustis, K. Papagiannaki, S. V. Krishnamurthy, M. Faloutsos, and V. Mhatre, "MDG: measurement-driven guidelines for 802.11 WLAN design," in *Proc. of the 13th annual ACM international conference on Mobile computing and networking*. ACM, 2007, pp. 254–265.

[4] "Aruba networks," http://www.arubanetworks.com/.

[5] "Cisco meraki," https://meraki.cisco.com/.

[6] R. Riggio, C. Sengul, L. Suresh, J. Schulz-Zander, and A. Feldmann, "Thor: Energy programmable wifi networks," in *Proc. of IEEE INFOCOM*, 2013.

[7] J. Lorincz, A. Capone, and M. Bogarelli, "Energy savings in wireless access networks through optimized network management," in *Wireless Pervasive Computing (ISWPC), 2010 5th IEEE International Symposium on*, pp. 449–454.

[8] A. Mishra, S. Banerjee, and W. Arbaugh, "Weighted coloring based channel assignment for wlans," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 3, pp. 19–31, 2005.

[9] Y. Lee, K. Kim, and Y. Choi, "Optimization of ap placement and channel assignment in wireless lans," in *Local Computer Networks. Proceedings. LCN 2002. 27th Annual IEEE Conference on*, pp. 831–836.

[10] R. C. Rodrigues, G. R. Mateus, and A. A. F. Loureiro, "On the design and capacity planning of a wireless local area network," in *Network Operations and Management Symposium, NOMS. 2000 IEEE/IFIP*, pp. 335–348.

[11] M. J. Neely, E. Modiano, and C.-P. Li, "Fairness and optimal stochastic control for heterogeneous networks," *Networking, IEEE/ACM Transactions on*, vol. 16, no. 2, pp. 396–409, 2008.

[12] S. Sakai, M. Togasaki, and K. Yamazaki, "A note on greedy algorithms for the maximum weighted independent set problem," *Discrete Applied Mathematics*, vol. 126, no. 2, pp. 313–322, 2003.

[13] V. Chvatal, "A greedy heuristic for the set-covering problem," *Mathematics of operations research*, vol. 4, no. 3, pp. 233–235, 1979.

[14] Z. Zheng, L. X. Cai, M. Dong, X. Shen, and H. V. Poor, "Constrained energy-aware ap placement with rate adaptation in wlan mesh networks," in *Global Telecommunications Conference (GLOBECOM 2011), IEEE.*

[15] R. Murty, J. Padhye, R. Chandra, A. Wolman, and B. Zill, "Designing high performance enterprise wi-fi networks." in *NSDI*, vol. 8, 2008, pp. 73–88.

[16] L. Suresh, J. Schulz-Zander, R. Merz, A. Feldmann, and T. Vazao, "Towards programmable enterprise WLANs with odin," in *Proc. of the first workshop on Hot topics in software defined networks*. ACM, 2012, pp. 115–120.

[17] K.-K. Yap, M. Kobayashi, R. Sherwood, T.-Y. Huang, M. Chan, N. Handigol, and N. McKeown, "Openroads: Empowering research in mobile networks," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 1, pp. 125–126, 2010.

[18] L. Jiang, S. Parekh, and J. Walrand, "Base station association game in multi-cell wireless networks (special paper)," pp. 1616–1621, 2008.

[19] S. Chiaravalloti, F. Idzikowski, and L. Budzisz, "Power consumption of wlan network elements," *Tech. Univ. Berlin, Tech. Rep. TKN-11-002*, 2011.

[20] Y. Yao, L. Huang, A. Sharma, L. Golubchik, and M. Neely, "Data centers power reduction: A two time scale approach for delay tolerant workloads," in *INFOCOM, 2012 Proceedings IEEE*, pp. 1431–1439.

[21] J. Kwak, O. Choi, S. Chong, and P. Mohapatra, "Dynamic speed scaling for energy minimization in delay-tolerant smartphone applications," in *INFOCOM, 2014 Proceedings IEEE*, pp. 2292–2300.

[22] *Extended Manuscript*, 2015. [Online]. Available: https://www.dropbox.com/s/qwksa3rvlbk82f6/ExtendedSubmissionInfocom2016.pdf?dl=0

[23] M. Afanasyev, T. Chen, G. M. Voelker, and A. C. Snoeren, "Usage patterns in an urban wifi network," *Networking, IEEE/ACM Transactions on*, vol. 18, no. 5, pp. 1359–1372, 2010.

[24] Y. Zeng, P. H. Pathak, and P. Mohapatra, "A first look at 802.11 ac in action: Energy efficiency and interference characterization," in *Networking Conference, 2014 IFIP*. IEEE, 2014, pp. 1–9.

[25] "Floodlight project," http://www.projectfloodlight.org/.

[26] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.

[27] "Power manager," http://www.powermanager.co.kr/en/.