Joint Access Point Placement and Power-Channel-Resource-Unit Assignment for 802.11ax-Based Dense WiFi with QoS Requirements

Shuwei Qiu, Xiaowen Chu, Yiu-Wing Leung, Joseph Kee Yin Ng
Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Kowloon, Hong Kong
{csswqiu, chxw, ywleung, jng}@comp.hkbu.edu.hk

Abstract—IEEE 802.11ax is a promising standard for the next-generation WiFi network, which uses orthogonal frequency division multiple access (OFDMA) to segregate the wireless spectrum into time-frequency resource units (RUs). In this paper, we aim at designing an 802.11ax-based dense WiFi network to provide WiFi services to a large number of users within a given area with the following objectives: (1) to minimize the number of access points (APs); (2) to fulfill the users’ throughput requirement; and (3) to be resistant to AP failures. We formulate the above into a joint AP placement and power-channel-RU assignment optimization problem, which is NP-hard. To tackle this problem, we first derive an analytical model to estimate each user’s throughput under the mechanism of OFDMA and a widely used interference model. We then design a heuristic algorithm to find high-quality solutions with polynomial time complexity. Simulation results show that our algorithm can achieve the optimal performance for a small area of 50 × 50 m² while we cannot find the optimal solution through an exhaustive search, our algorithm can reduce the number of APs by 32 ~ 55% as compared to the random and Greedy solutions.

Index Terms—IEEE 802.11ax, AP placement, quality of service, fault tolerance, resource assignment, dense WiFi network

I. INTRODUCTION

The IEEE 802.11ax-based dense WiFi network has attracted more and more attention from the industry and academia [1]. In dense WiFi scenarios, many users are gathered in a region, which creates a great demand on WiFi services, like upload/download videos to/from the network. In this case, many access points (APs) are required and the distance between adjacent APs is usually very close [1]. In a traditional 802.11-based dense WiFi setup, sufficient bandwidth does not necessarily translate into high throughput [2]. To improve the user experience in dense WiFi scenarios, 802.11ax [3], which acts as the next generation WiFi standard, has been investigated. It supports orthogonal frequency division multiple access (OFDMA), in which subcarriers in a channel are divided into groups which are called resource units (RUS) [3]. By strategically assigning RUs to stations, stations can transmit data simultaneously. In addition, it supports both the 2.4 and 5 GHz bands, which means that we have more non-overlapping channels to choose from to reduce the interference between neighboring APs. In short, deploying 802.11ax-based dense WiFi network is both crucial and urgent. There are two main factors that affect the network performance. The first one is the AP placement. The second one is resource (such as power, channel, and RU, etc.) assignment for the APs/stations. Furthermore, users demand continuous WiFi services even under AP’s failures [4]. Unfortunately, there is little research on joint AP placement and resource assignment for 802.11ax-based dense WiFi network with quality of service (QoS) guarantees. This is why we try to address this problem.

We consider a given region with many potential users at known locations (e.g., in a stadium, each spectator is assigned a fixed seat). Our problem can be described as follows. Given a set of AP possible candidate locations and a set of stations with known locations, find out the minimum number of APs and their locations, under the joint design of AP placement and power-channel-RU assignment, to deploy an 802.11ax-based dense WiFi network that fulfills the following two QoS requirements: 1) Fault tolerance requirement. That is, when n (n = 0, 1, 2, ...) APs fail simultaneously, stations associated with the failed APs can still re-associate with the remaining APs to obtain the WiFi service. 2) User satisfaction ratio (USR) requirement, which is a new concept introduced in our paper. That is, we ensure that the throughput of at least \( \beta \% \) (\( 0 \leq \beta \leq 100 \)) of the stations is no less than \( \rho_H \), and the throughput of \( (100 - \beta) \% \) of the stations is no less than \( \rho_L \), where \( \rho_L \) and \( \rho_H \) represent two throughput thresholds which can be obtained beforehand by historical data with \( \rho_L < \rho_H \). We assume that \( \rho_L \) is the minimum throughput that users can accept while \( \rho_H \) is the throughput that users are satisfied with. The reason why we introduce the user satisfaction ratio requirement is that in dense WiFi scenarios, it is expensive to satisfy all users. In fact, the locations of some stations may be far away from the existing APs. If we have to satisfy them, we may need to add many APs to the network which is not cost effective. But instead, we just ensure that the throughput of the far-away stations is no less than \( \rho_L \).
only meets the basic needs but also saves cost in providing the WiFi service. Hence, the contributions of our research include the followings.

1) **New Problem:** We deploy an IEEE 802.11ax-based dense WiFi network in a given region with many potential users by jointly optimizing AP placement and power-channel-RU assignment. We formulate an optimization problem that minimizes the number of APs subject to the fault tolerance constraints and the user satisfaction ratio requirements. As the sub-problems of our problem are NP-hard, our problem is therefore NP-hard as well. The main differences between our work and the existing ones are summarized in Table I, which shows that our problem is quite different from the others.

2) **New Solution:** According to the mechanism of OFDMA and a widely used interference model, we design efficient power adjustment, channel assignment, and RU assignment methods, based on which we derive the throughput of the stations. We propose a heuristic algorithm with polynomial time complexity to solve the optimization problem. We conduct extensive simulations with various parameter settings. Our simulation results demonstrate that our heuristic algorithm gives high quality solutions for the optimization problem.

The remainder of this paper is organized as follows. Section II reviews the related work. Section III formulates the problem and Section IV describes how we obtain the throughput of stations. Our algorithm is presented in Section V and Section VI describes our simulations and presents the performance of our algorithm. And lastly, Section VII concludes this paper.

### II. RELATED WORK

#### A. Unique Features of the IEEE 802.11ax Standard

The first two IEEE 802.11ax drafts, D1.0 and D2.0, were released in 2016 and 2017 [5], respectively. The latest one, D3.0, was released in 2018 [3]. IEEE 802.11ax aims to improve the throughput by a factor of at least 4 [6] as compared to 802.11n/ac in dense scenarios [7]. IEEE 802.11ax has the following features. 1) The use of OFDMA [8] [9] which employs multiple subcarriers that are divided into multiple RUs where RUs are allocated to stations for supporting simultaneous transmission [10] [11]. 2) The use of Downlink/Uplink multi-user multiple-input multiple-output (DL/UL MU MIMO) [12] [13], which improves the throughput by using multiple spatial streams. 3) The use of trigger frame such that AP can coordinate the concurrent transmissions of stations, and that it indicates the number of spatial streams and/or the RU size of each station. 4) The use of random access protocol [14] [15] such that when an AP senses that some stations are going to transmit, but does not know which stations they are, it can assign some RUs for multiple stations to transmit through a random access mechanism.

#### B. IEEE 802.11ax-Based Dense WiFi Network

IEEE 802.11ax-based dense WiFi network has been attracting researchers’ attention recently. Bellalta et al. [1] present some of the network-level functionalities that are required to improve the user experience in dense WiFi scenarios. Deng et al. [2] point out that IEEE 802.11ax will fuel the future intelligent information infrastructure for big data transfer and various mobile applications. Deng et al. [16] discuss the challenges for IEEE 802.11ax in the design of physical layer and medium access control (MAC) sub-layer. Furthermore, they present the expected features on the MAC protocol design to provide better QoS support in the IEEE 802.11ax-based dense WiFi network [17]. Afaqui et al. [18] disclose advanced technological enhancements presented in IEEE 802.11ax to improve the user throughput within a dense WiFi network. All the above results show that IEEE 802.11ax-based dense WiFi network will become popular in the near future.

#### C. AP Placement and Fault Tolerance

AP placement and fault tolerance have been intensively investigated. Ling et al. [19] jointly solve the two problems of AP placement and channel assignment for providing better network services. Zheng et al. [20] study the AP placement problem aiming to minimize the number of APs being used. In [21], an AP placement problem is formulated, whose objective is to determine the optimal placement of APs. Zhang et al. [22] address the AP placement problem that AP can be equipped with multiple radios to minimize the total cost of all APs. Zhang [23] et al. present an optimization framework of AP placement, whose aim is to maximize the signal coverage. Kiran [24] et al. focus on the optimization of the AP placement.
to maximize the coverage by optimizing the power allocation. Audhya [25] et al. optimally place the APs in an ultra-dense 5G network to cover a given region. In addition, Zhou et al. [4] study the problem of enhancing the fault tolerance of a WiFi network. They consider the situation that when an AP fails, the stations it serves shall switch to other APs to obtain acceptable services. Liu et al. [26] propose a self-healing scheme to provide a continuous service for users in ultra-dense network. Moreover, Lee et al. [27] propose a resource allocation algorithm to overcome the unforeseen AP failures. These works, however, do not consider the joint design of AP placement and power-channel-RU assignment.

III. PROBLEM FORMULATION

The symbols used in our model are shown in Table II.

<table>
<thead>
<tr>
<th>Symb</th>
<th>Meaning</th>
<th>Symb</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>The set of APs</td>
<td>$A_f$</td>
<td>The set of fault APs</td>
</tr>
<tr>
<td>$S$</td>
<td>The set of stations (STA)</td>
<td>$B$</td>
<td>The set of channel widths</td>
</tr>
<tr>
<td>$P$</td>
<td>The set of power levels</td>
<td>$C$</td>
<td>The set of channels</td>
</tr>
<tr>
<td>$K$</td>
<td>Subcarrier number set</td>
<td>$R_i$</td>
<td>The data rate of STA $i$</td>
</tr>
<tr>
<td>$\delta_i$</td>
<td>The throughput of STA $i$</td>
<td>$\delta_i^{\text{ref}}$</td>
<td>$1$ if $\delta_i \geq \rho_H$, $0$ otherwise</td>
</tr>
<tr>
<td>$I_{i,j}$</td>
<td>The interference range between APs $i$ and $j$</td>
<td>$\delta_i^{(L)}$</td>
<td>$1$ if $\rho_H &gt; \delta_i \geq \rho_L$, $0$ otherwise</td>
</tr>
<tr>
<td>$\Omega$</td>
<td>The set of AP candidate locations</td>
<td>$N(i)$</td>
<td>The set of neighboring APs of AP $i$</td>
</tr>
<tr>
<td>$A(i)$</td>
<td>The set of APs that cover STA $i$</td>
<td>$S(i)$</td>
<td>The set of STAs that associate with AP $i$</td>
</tr>
</tbody>
</table>

A. The Network Model

We assume that a set of possible AP locations, denoted by $\Omega$, which is given beforehand. That is, the target region is divided into multiple cells and APs can be placed only at the center of them. Within one cell location, zero, one, or more APs can be placed according to the density of the stations (STAs). Our network consists of three kinds of devices: network controller, APs, and STAs. The network controller is responsible for network management and coordination. APs do not need to backoff to beamforming before the coordination under the coordination of the network controller. Denote $S$ and $A$ to be the set of STAs and the set of APs, respectively. Any STA $i \in S$ can only associate with one AP $j \in A$ during a period of time. When a set of APs, $A_f$ ($|A_f| < |A|$), fail, the STAs which are previously associated with the failed APs can re-associate with other APs $\in A \setminus A_f$ to obtain the WiFi service. We adopt both the 2.4 and 5 GHz bands, in which a channel with $b$ MHz, ($b \in B = \{20, 40, 80, 160\}$ MHz [3]) bandwidth can be assigned to an AP. In our model, the OFDMA physical layer is adopted. Instead of the whole channel, the RUs are assigned to STAs for simultaneous transmissions. Each AP can only be assigned a channel in a given channel set, denoted by $C$. Each STA can only be assigned a $j$-tone RU [3] ($j \in K$, where $K$ is the set of subcarrier numbers). In addition, each AP is assigned a power level within a given power level set, denoted by $P$. The power level of each STA is the same as that of its AP [21]. The frame exchange procedure is shown in Fig. 1 [17], where TXOP, SIFS, M-BA, and OFDMA-BA represent the transmission opportunity, the short interframe space, the multistation block ACK, and the OFDMA block ACK, respectively [3]. Under OFDMA, STAs start to transmit Uplink physical layer protocol data unit (PPDU) to its AP only after receiving a trigger frame (TF). The STAs reply an OFDMA-BA frame to the AP after receiving the Downlink PPDU.

![Frame exchange procedure between an AP and its STAs](image)

B. The Interference Model

Let $l_{i,j}$ denote the link between nodes $i$ and $j$ (here, a node refers to an AP or a STA). To let node $i$ receive frame properly from $j$ over link $l_{i,j}$, the received signal strength (RSS) at node $i$ must be no less than the frame decoding threshold $\theta_D$ [22]. In this case, we say that node $i$ is within the transmission range of $j$ and vice versa. In addition, node $i$ is said to be interfered by $j$ (here, nodes $i$ and $j$ are on different links whose channels overlap with each other) if the signal strength received by node $i$ from $j$ is greater than or equal to the interference signal strength threshold $\theta_I$ [22]. In this case, we say that node $i$ is within the interference range of $j$ and vice versa. Usually, $\theta_D > \theta_I$. To obtain the communication ranges and the interference ranges of APs, we resort to the following path loss model [29]: $RSS = P_j + G_{TX,i} - P_{loss} + G_{RX}$, where $P_{loss} = P_{ref} + 10 \log(d^\eta) + \chi$. Here, $RSS$ is the received signal strength at the receiver; $d$ is the distance between the sender and the receiver; $P_j$ is the power level of sender node $j$; $G_{TX,i}$ and $G_{RX}$ are the antenna gains of the sender and the receiver respectively; $P_{ref}$ is the path loss at a reference distance (which is usually 1 m); $\eta$ is the path-loss exponent; and $\chi$ is the standard deviation associated with the degree of shadow fading. Thus, $d = \sqrt{\frac{RSS - P_{ref} - \chi - G_{RX}}{10 \log(\eta)}}$. Let $r_j$ and $\gamma_j$ denote the communication range and the interference range of node $j$, respectively. We have $d = r_j$ if $RSS = \theta_D$ and $d = \gamma_j$ if $RSS = \theta_I$. Next, we introduce the interference model [30]. Let $l_{i,x}$ and $l_{j,y}$ denote the links between AP $i$ and STA $x$, and AP $j$ and STA $y$, respectively. Let $d_{i,x}$ and $d_{j,y}$ denote the distance between AP $i$ and STA $x$, and AP $j$ and STA $y$, respectively. Let $\gamma_x$ and $\gamma_y$ denote the interference range of STA $x$ and STA $y$, respectively. Fig. 2 depicts the interference ranges of links $l_{i,x}$ (the region enclosed by dotted line) and $l_{j,y}$ (the region enclosed by solid line).

Denote $S(i)$ and $S(j)$ the sets of STAs that are associated with APs $i$ and $j$, respectively. According to Fig. 2, we define the interference range between APs $i$ and $j$ as
I then links C. The Optimization Problem cannot transmit simultaneously [22].

maximum level (which will be adjusted later) to cover as many APs from which their signals can cover STA i. If not, we add APs to set A until C2 and C3 can be met. Thus, A(i) = \{AP j|d_{i,j} \leq r_{j}\}, i \in S, j \in A.

Then, we associate STA i to the AP in A(i) whose signal strength received by STA i is the strongest. After the STAs-APs association, the set of STAs associated with AP j, S(j) (j \in A), can also be obtained.

B. Power Adjustment

As mentioned earlier, the power level of each AP is initialized to the maximum in P. But higher power level leads to larger interference range, so we need to downward adjust the power levels of APs to reduce their interference among each other. Denote \( p_{j}^{(q)} \) the communication range of AP j with power level \( p_{j} \). We have \( r_{j}^{(1)} < r_{j}^{(2)} < \ldots < r_{j}^{(|P|)} \) under the assumption of \( p_{1} < p_{2} < \ldots < p_{|P|} \). After the STAs-APs association, we know the maximum distance between AP j and its STAs, \( \max_{x \in \{1,\ldots,|S|\}} \{d_{i,j}\} \). Then the power level of AP j can be adjusted as follows:

\[
P_{j} = \begin{cases} \begin{align*}
 p_{q}, & \text{if } \max_{x \in \{1,\ldots,|S|\}} \{d_{i,j}\} \in (r_{j}^{(q-1)}, r_{j}^{(q)}), q \in [2,|P|]; \\
p_{1}, & \text{if } \max_{x \in \{1,\ldots,|S|\}} \{d_{i,j}\} \leq r_{j}^{(1)}.
\end{align*} \end{cases}
\]

(2)

With \( P_{j} (j \in A) \) in hands, we can further obtain the interference range between APs i and j, \( I_{i,j}(i \neq j) \), which can help to assign channels to the APs.

C. Channel Assignment and Power Re-adjustment

We use both the 2.4 and 5 GHz [31] bands for transmissions. The channels used in our model are shown in Fig. 3. Thus, the channel set \( C = \{1, 2, \ldots, 19\} \). From Fig. 3 we observe that channel 12 overlaps with channels 2 and 3. Hence, we define two overlapping channel sets (OCSs) in the 2.4 GHz band, denoted by \( \Gamma_{1} = \{2, 12\} \) and \( \Gamma_{2} = \{3, 12\} \), respectively. Similarly, we define eight OCSs in the 5 GHz band, denoted by \( \Gamma_{3} = \{4, 13, 17, 19\} \), \( \Gamma_{4} = \{5, 13, 17, 19\} \), \( \ldots \), \( \Gamma_{10} = \{11, 16, 18, 19\} \). We call channels 1 to 11 primary channels.

Denote the set of neighboring APs of AP i by N(i), which is defined as N(i) = \{AP j|d_{i,j} < r_{j}\}, i, j \in A, i \neq j, where \( D_{i,j} \) denotes the distance between APs i and j. For any AP i, if AP j \in N(i) and the non-overlapping channels are enough, then we assign a channel which does not overlap with AP j’s channel to AP i. When the non-overlapping channels are not enough, the channel assigned to AP i may overlap with AP

\[
I_{i,j} = \max_{x \in S(j)} \{d_{i,x}\} + \max_{y \in S(j)} \{d_{j,y}\} + \max_{y \in S(j)} \{d_{i,y}\}.
\]

Fig. 2. Interference range between APs i and j [30].

\[
\begin{align*}
\text{Procedure 1: Obtaining the throughput of STAs} \\
\text{Input: } A, S, P, C, \ldots \\
\text{Output: } \delta_{i} (i \in S) \\
\text{Step 1. STAs-APs association.} \\
\text{Step 2. Power adjustment for the APs.} \\
\text{Step 3. Channel assignment and power re-adjustment for the APs.} \\
\text{Step 4. RU assignment for the STAs.} \\
\text{Step 5. Obtaining the data rate of STAs.} \\
\text{Step 6. Calculating the throughput of STAs.}
\end{align*}
\]

\[
\begin{align*}
\text{C1: } & \sum_{j=1}^{A} a_{i,j} = 1, i \in S; \\
\text{C2: } & \sum_{i=1}^{S} \delta_{i}^{H} \geq |S| \times \beta\%; \\
\text{C3: } & \sum_{i=1}^{S} \delta_{i}^{L} + \delta_{i}^{H} = |S|.
\end{align*}
\]

\[
\begin{align*}
\text{D. The General Framework of Our Solution} \\
\text{To solve (1), } \delta_{i} (i \in S) \text{ should be obtained first (see Section IV). For a given set } S \text{ and an AP placement scheme } A \text{ that satisfies C1 (if } A \text{ cannot satisfy C1, we add APs to set } A \text{ until it satisfies C1), we adopt Procedure 1 to obtain } \delta_{i}. \text{ After } \delta_{i} (i \in S) \text{ is obtained, we check whether set } A \text{ fulfills C2 and C3. If not, we add APs to set } A \text{ until C2 and C3 can be met.}
\end{align*}
\]

IV. THROUGHPUT OF STATIONS

A. STAs-APs Association

To do the STAs-APs association, we first obtain the set of APs from which their signals can cover STA i, denoted by A(i). We initialize the power level of each AP in P to its maximum level (which will be adjusted later) to cover as many STAs as possible. That is, \( P_{j} = \max_{p \in P}\{p_{q}\}, j \in A \), where \( p_{q} \) denotes the q-th power level in P. If the distance between STA i and AP j, \( d_{i,j} \), is less than or equal to the communication range of AP j, \( r_{j} \), then the signal emitted from AP j can cover STA i. Thus, \( A(i) = \{AP j|d_{i,j} \leq r_{j}\}, i \in S, j \in A. \)

Then, we associate STA i to the AP in A(i) whose signal strength received by STA i is the strongest. After the STAs-APs association, the set of STAs associated with AP j, S(j) (j \in A), can also be obtained.

B. Power Adjustment

As mentioned earlier, the power level of each AP is initialized to the maximum in P. But higher power level leads to larger interference range, so we need to downward adjust the power levels of APs to reduce their interference among each other. Denote \( p_{j}^{(q)} \) the communication range of AP j with power level \( p_{j} \). We have \( r_{j}^{(1)} < r_{j}^{(2)} < \ldots < r_{j}^{(|P|)} \) under the assumption of \( p_{1} < p_{2} < \ldots < p_{|P|} \). After the STAs-APs association, we know the maximum distance between AP j and its STAs, \( \max_{x \in \{1,\ldots,|S|\}} \{d_{i,j}\} \). Then the power level of AP j can be adjusted as follows:

\[
P_{j} = \begin{cases} \begin{align*}
 p_{q}, & \text{if } \max_{x \in \{1,\ldots,|S|\}} \{d_{i,j}\} \in (r_{j}^{(q-1)}, r_{j}^{(q)}), q \in [2,|P|]; \\
p_{1}, & \text{if } \max_{x \in \{1,\ldots,|S|\}} \{d_{i,j}\} \leq r_{j}^{(1)}.
\end{align*} \end{cases}
\]

(2)

With \( P_{j} (j \in A) \) in hands, we can further obtain the interference range between APs i and j, \( I_{i,j}(i \neq j) \), which can help to assign channels to the APs.

C. Channel Assignment and Power Re-adjustment

We use both the 2.4 and 5 GHz [31] bands for transmissions. The channels used in our model are shown in Fig. 3. Thus,
j’s channel. In this case, we try to reduce the total interference degree between AP i and its neighbors (see next paragraph). Obviously, if AP j \notin N(i), then the channel assigned to AP i can be the same as that being assigned to AP j.

We introduce a channel conflict indicator (CCI) to measure the interference degree of APs. Let CCI_i denote the interference degree of AP i, which is defined as: the number of AP i’s neighbors whose channels belong to the same OCS as the channel of AP i. A CCI graph is presented as shown in Fig. 4 to help us assign the channels. In Fig. 4, the circle represents the AP whose channel number is presented at the center. The two APs connected by an edge are neighbors to each other. For example, in Fig. 4(a), we want to assign a channel to AP 1 whose channel number is initialized as 0. Suppose that no other channels can be used. Thus, we can choose one only from \{5, 13, 14, 17\} for AP 1. Notice that 5, 13, 17 \in \Gamma_4, we assign 14 to AP 1 (see Fig. 4(b)). The reason is that any one of 5, 13, 17, such as 17, assigning to AP 1 increases the CCI values of AP 1 and its neighbors significantly (see Fig. 4(c)), which leads to a higher degree of interference.

The channel assignment algorithm is shown in Algorithm 1. At the beginning, the channel numbers and CCI values of all APs are initialized to 0. First, we generate a channel assignment queue Q_\alpha by sorting the APs in descending order according to the number of STAs served by each AP. Then we assign channels to APs one by one according to the AP order in Q_\alpha (we construct corresponding CCI graph at the same time). That is, the APs with more STAs are given higher priority to be assigned to the channels. In Algorithm 1, c_{i,j} is equal to 1 if channel j (j \in C) is assigned to AP i (i \in A), and 0 otherwise. Lines 8 \sim 9 mean that we assign the first channel (i.e., the primary channel) in set C^* to AP i when the non-overlapping channels are enough (i.e., C^* \neq \emptyset). Lines 10 \sim 13 mean that when the non-overlapping channels are not enough, AP i is always assigned the channel with the minimum increase of CCI values of AP i and its neighbors. Lines 14 \sim 16 update the channels that have been assigned to the APs to channels with wider bandwidth.

After the channel assignment, we re-adjust the power level of APs to increase the RSS of STAs (i.e., to increase their data rate). For each AP i whose power level p_q is lower than p_{[i]}, we adjust its power level from p_q to p_{q+1} (q \in [1, |P| - 1]), and then judge whether the signals emitted from AP i with power level p_{q+1} interfere with other basic service sets. If so, then we restore the power level to its original value (i.e., p_q) and stop; otherwise, we continue to adjust its power level from p_{q+1} to p_{q+2} until the power level is equal to p_{[i]}.

D. RU Assignment

There are seven types of RUs defined in IEEE 802.11ax [3], that is, the subcarrier (i.e., tone) number set K = \{26, 52, 106, 242, 484, 996, 2 \times 996\}. The maximum number of k-tone RUs (k \in K) for each channel width are shown in Fig. 5(a), where b_1 = 20, b_2 = 40, b_3 = 80, and b_4 = 160 (MHz) [3]. It implies that up to 9 STAs in 20 MHz, 18 STAs in 40 MHz, 37 STAs in 80 MHz, and 74 STAs in 160 MHz channel width are supported in an OFDMA transmission. Fig. 5(b) shows the RU locations in a 20 MHz PPDU [3]. The RU locations in a 40, 80, or 160 MHz PPDU can be found in [3]. As shown in Fig. 5, the maximum number of k-tone RUs (k \in K) is determined by the channel width. Hence, when assigning RUs to AP i’s STAs, we consider the number of STAs served by AP i, |S(i)|, as well as the channel width of AP i. We mainly focus on the following two aspects: 1) how to utilize the bandwidth of AP i’s channel as much as possible; 2) how to balance the data rate of STAs in S(i) as much as possible. For aspect one, we define m_b RU sets, RU_{b,m} (b \in B, m = 1, 2, ..., m_b, where m_b is the maximum number of 26-tone RUs in b MHz channel width), for RU assignment and guarantee that the total bandwidth of the RUs in set RU_{b,m}
is as close to \( b \) as possible. For example, according to Fig. 5(b), we define \( RU_{20,1} = \{242\} \) for 1 STA, \( RU_{20,2} = \{106, 106\} \) for 2 STAs, \( RU_{20,3} = \{26, 106, 106\} \) for 3 STAs, \( ... \), \( RU_{20,9} = \{26, 26, 26, 26, 26, 26, 26, 26, 26\} \) for 9 STAs. For **aspect two**, we assign larger-size RUs to STAs that are farther away from AP and smaller-size RUs to STAs that are nearer to AP \( i \). For any AP \( i \) with channel width \( b \), we adopt the following steps to assign RUs to its STAs.

**Step 1.** We divide \( |S(i)| \) by \( m_b \) to get the quotient of \( |S(i)|/m_b \) with the remainder of \( \text{rem} \).

**Step 2.** We divide AP \( i \)'s STAs into \( |S(i)/m_b| + 1 \) groups. The \( x \)-th group \( (x = 1, 2, ..., |S(i)/m_b|) \) contains \( m_b \) STAs and the \((|S(i)/m_b| + 1)\)-th group contains \( \text{rem} \) STAs.

**Step 3.** We assign the RUs in set \( RU_{b,m_b} \) to the STAs in the \( x \)-th group \( (x = 1, 2, ..., |S(i)/m_b|) \), and assign the RUs in set \( RU_{b,\text{rem}} \) to the STAs in the \((|S(i)/m_b| + 1)\)-th group.

These \( |S(i)/m_b| + 1 \) groups of AP \( i \)'s STAs exchange frames with AP \( i \) in turn.

**E. Data Rate of STAs**

We obtain the data rate of STAs according to the RSS and the RU of STAs. From [3], we can get the correspondence between the receiver minimum input level sensitivity and data rate, that is, \( (MS_b_1, \sigma_{k_1}), (MS_b_2, \sigma_{k_2}), \ldots, (MS_b_X, \sigma_{k_X}) \), where \( MS_b_x (b \in B, x = 1, 2, \ldots, X) \) denotes the \( x \)-th minimum sensitivity (MS) in \( b \) MHz channel width; \( \sigma_{k_X} \) denotes the \( x \)-th data rate for \( k \)-tone RU \((k \in K)\). Here, \( MS_{b_1} < MS_{b_2} < \ldots < MS_{b_X} \) and \( \sigma_{k_1} < \sigma_{k_2} < \ldots < \sigma_{k_X} \). The data rate of an UL (or DL) from STA \( i \) to AP \( j \) (or from AP \( j \) to STA \( i \)) depends on the RSS at AP \( j \) (or STA \( i \)) [22]. Denote RSS\(_i\), the signal strength received by AP \( j \) from STA \( i \), and \( R_t \) the data rate of STA \( i \) for UL data traffic. Thus, we have \( R_t = \sigma_{k,q} \) if \( MS_{b,q} \leq \text{RSS}_i < MS_{b,q+1} \), \( q \in \{1, X-1\} \); and \( R_t = \sigma_{k,X} \) if \( MS_{b,X} \leq \text{RSS}_i \). Since we assume that the power level of STA \( i \)'s AP is the same as that of STA \( i \) [21], the data rate of STA \( i \) for DL data traffic is the same as \( R_t \).

**F. Throughput of STAs**

For any AP \( i \) with channel width \( b \), there are \( |S(i)| \) STAs associated with it. Thus, it needs

\[
M = \begin{cases} 
|S(i)|/m_b, & \text{if } |S(i)| = zm_b, z \in \mathbb{Z}^+; \\
|S(i)|/m_b + 1, & \text{otherwise}, 
\end{cases}
\]

(3)

frame exchanges (including DL and UL traffics) to complete one communication round (i.e., each STA in \( S(i) \) completes one UL transmission and one DL reception).

Let \( t_{TF}, t_{SIFS}, t_{UL,PPDU}, t_{DL,PPDU}, \) and \( t_{OFDMA,BA} \) to denote the duration of TF, SIFS, UL PPDU, M-BA, DL PPDU, and OFDMA-BA, respectively. Denote the duration of an UL transmission by \( T_{UL} \), the duration of a DL reception by \( T_{DL} \). According to Fig. 1, we have \( T_{UL} = t_{TF} + 2t_{SIFS} + t_{UL,PPDU} + t_{M,BA} \) and \( T_{DL} = 2t_{SIFS} + t_{DL,PPDU} + t_{OFDMA,BA} \). Denote the duration of one communication round by \( T \), then \( T = (T_{UL} + T_{DL})M \).

Thus, the throughput of STA \( i \) associated with AP \( j \) is

\[
\delta_i = \frac{R_i(t_{UL,PPDU} + t_{DL,PPDU})}{T(CCI_j + 1)},
\]

(4)

where \( CCI_j + 1 \) means that AP \( j \) and its \( CCI_j \) neighbors interfere with each other. That is, they must be active in turn.

**V. ALGORITHM DESIGN AND ANALYSIS**

Since the sub-problems arise from our problem (such as the optimal AP placement etc.) are NP-hard [22], our problem is therefore NP-hard as well. Thus, we design a polynomial time heuristic algorithm to provide a solution for it.

**A. The Heuristic Algorithm**

Our heuristic algorithm consists of four stages (see below for technical details in each stage). The key operation in each stage is to check whether a given solution \( A \) is feasible. Thus, we first design Algorithm 2 for the feasibility test.

**Algorithm 2: Feasibility test for solution \( A \)**

- **Input**: \( A, S, A, P, A, n, \) etc.
- **Output**: an indicator \( I \) if \( A \) is feasible, then \( I = TRUE \); otherwise \( I = FALSE \).

\[
\begin{align*}
1 & \quad I \leftarrow TRUE; \quad A^* \leftarrow A; \\
2 & \quad \text{if } C1 \text{ can be met then} \\
3 & \quad \text{for each } A_j \in A \text{ do} \\
4 & \quad \quad A \leftarrow A^*; \\
5 & \quad \quad A \leftarrow A \setminus A_j; \\
6 & \quad \quad \text{Call Procedure 1 to obtain the throughput of STAs;} \\
7 & \quad \quad \text{if } C2 \text{ and } C3 \text{ cannot be met then} \\
8 & \quad \quad \quad I \leftarrow FALSE; \\
9 & \quad \text{Quit the loop;} \\
10 & \quad \text{else} \\
11 & \quad \quad I \leftarrow FALSE; \\
12 & \quad \text{Return } I;
\end{align*}
\]

**In stage one**, we use the Greedy algorithm as shown in Algorithm 3 to iteratively place AP at the location around which the density of uncovered STAs is the highest until all STAs are covered and \( C1 \sim C3 \) in (1) can be met to get an initial solution \( A_1 \).

**Algorithm 3: Stage one-constructing an initial set of APs**

- **Input**: \( S, \Omega, A, P, n, \) etc.
- **Output**: \( A_1 \).

\[
\begin{align*}
1 & \quad A_1 \leftarrow \emptyset; \quad i \leftarrow 0; \\
2 & \quad \text{repeat} \\
3 & \quad \quad i \leftarrow i + 1; \\
4 & \quad \quad \text{Place AP } i \text{ at the location around which the density of uncovered STAs is the highest;} \\
5 & \quad \quad A_1 \leftarrow A_1 \cup \{\text{AP } i\}; \\
6 & \quad \quad \text{Call Algorithm 2 to test the feasibility of } A_1; \\
7 & \quad \quad \text{if the output of algorithm 2 is } FALSE \text{ then} \\
8 & \quad \quad \quad \text{Return to the } \text{repeat} \text{ statement;} \\
9 & \quad \quad \text{until the output of algorithm 2 is } TRUE; \\
10 & \quad \text{Return } A_1;
\end{align*}
\]

**In stage two**, we try to iteratively remove redundant APs in \( A_1 \) one by one in a predefined order. That is, in each iteration, we generate an AP queue \( Q_b \) by sorting the APs in ascending order according to the number of STAs served by each AP. Then we try to remove the first AP in \( Q_b \). If it cannot be removed (i.e., once it is removed, \( C1, C2 \) or \( C3 \) can not be met), then we keep it and try to remove the next one in \( Q_b \);
otherwise, we remove it and start the next iteration. The reason why we always try to remove the AP at the head of \(Q_d\) first in each iteration is that the AP with the least number of STAs is more likely to be removed. The operations of stage two are shown in Algorithm 4.

**Algorithm 4**: Stage two-removing the redundant APs

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A_2)</td>
<td>(A_2)</td>
</tr>
<tr>
<td>(A_2 \leftarrow A_1);</td>
<td></td>
</tr>
<tr>
<td>repeat</td>
<td></td>
</tr>
<tr>
<td>3 Generate queue (Q_d):</td>
<td></td>
</tr>
<tr>
<td>for each AP (i \in Q_d) do</td>
<td></td>
</tr>
<tr>
<td>(A_2 \leftarrow A_2 \setminus {\text{AP } i});</td>
<td></td>
</tr>
<tr>
<td>Call Algorithm 2 to test the feasibility of (A_2);</td>
<td></td>
</tr>
<tr>
<td>if the output of algorithm 2 is FALSE then</td>
<td></td>
</tr>
<tr>
<td>(A_2 \leftarrow A_2 \cup {\text{AP } i});</td>
<td></td>
</tr>
<tr>
<td>else</td>
<td></td>
</tr>
<tr>
<td>[ Return to the repeat statement;</td>
<td></td>
</tr>
<tr>
<td>until (i =</td>
<td>Q_d</td>
</tr>
<tr>
<td>Return (A_2);</td>
<td></td>
</tr>
</tbody>
</table>

In stage three, we iteratively replace two nearby APs in set \(A_2\) by one. There are two main tasks in each iteration: 1) finding the pair of APs with the shortest distance; and 2) trying to replace the pair of APs by a new one with a feasible location in \(\Omega\). The main steps in each iteration are as follows:

**Step 1.** We generate \(\binom{|A_2|}{2}\) pairs of APs and calculate the distance of each pair of APs.

**Step 2.** We generate a replacement queue \(Q_c[i] = \{\text{AP } i_1, \text{AP } i_2\}\), where APs \(i_1\) and \(i_2\) represent the two APs of the \(i\)-th pair of APs \((i = 1, 2, \ldots, \binom{|A_2|}{2})\), by sorting the \(\binom{|A_2|}{2}\) pairs of APs in ascending order according to the distance of each pair of APs.

**Step 3.** We iteratively try to replace the pair of APs at the head of \(Q_c\) by one until all pairs of APs have been tried.

The operations of stage three are shown in Algorithm 5.

**Algorithm 5**: Stage three-replacing two nearby APs by one

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A_3)</td>
<td>(A_3)</td>
</tr>
<tr>
<td>(A_3 \leftarrow A_2);</td>
<td></td>
</tr>
<tr>
<td>repeat</td>
<td></td>
</tr>
<tr>
<td>3 Generate queue (Q_c);</td>
<td></td>
</tr>
<tr>
<td>for each (Q_c[i] \in Q_c) do</td>
<td></td>
</tr>
<tr>
<td>(A_3 \leftarrow A_3 \setminus Q_c[i]);</td>
<td></td>
</tr>
<tr>
<td>for each location (g \in \Omega) do</td>
<td></td>
</tr>
<tr>
<td>Place a new AP at location (g);</td>
<td></td>
</tr>
<tr>
<td>(A_3 \leftarrow A_3 \cup {\text{the new AP}});</td>
<td></td>
</tr>
<tr>
<td>Call Algorithm 2 to test the feasibility of (A_3);</td>
<td></td>
</tr>
<tr>
<td>if the output of algorithm 2 is FALSE then</td>
<td></td>
</tr>
<tr>
<td>(A_3 \leftarrow A_3 \cup Q_c[i]);</td>
<td></td>
</tr>
<tr>
<td>(A_3 \leftarrow A_3 \setminus {\text{the new AP}});</td>
<td></td>
</tr>
<tr>
<td>else</td>
<td></td>
</tr>
<tr>
<td>[ Return to the repeat statement;</td>
<td></td>
</tr>
<tr>
<td>until (i =</td>
<td>Q_c</td>
</tr>
<tr>
<td>Return (A_3);</td>
<td></td>
</tr>
</tbody>
</table>

In stage four, we continue to try to reduce the number of APs by iteratively replacing three nearby APs in set \(A_3\) by two. The main steps in each iteration are as follows:

**Step 1.** We generate \(\binom{|A_3|}{3}\) groups of APs with each containing 3 APs.

**Step 2.** We calculate the distance between APs \(i_1\) and \(i_2\), \(D_{i_1,i_2}\); APs \(i_2\) and \(i_3\), \(D_{i_2,i_3}\); and APs \(i_1\) and \(i_3\), \(D_{i_1,i_3}\), respectively, where APs \(i_1\), \(i_2\), and \(i_3\) denote the three APs of the \(i\)-th group \((i = 1, 2, \ldots, \binom{|A_3|}{3})\).

**Step 3.** We calculate the distance between APs \(i_1\), \(i_2\), and \(i_3\), \(G_i\), which is defined by \(G_i = D_{i_1,i_2} + D_{i_2,i_3} + D_{i_1,i_3}\).

**Step 4.** We generate a replacement queue \(Q_d[i] = \{\text{AP } i_1, \text{AP } i_2, \text{AP } i_3\}\) by sorting the \(\binom{|A_3|}{3}\) groups of APs in ascending order according to the values of \(G_i\) \((i = 1, 2, \ldots, \binom{|A_3|}{3})\).

**Step 5.** We iteratively try to replace the group of APs at the head of \(Q_d\) by two until no group of APs can be replaced.

The operations of stage four are shown in Algorithm 6.

**Algorithm 6**: Stage four-replacing three nearby APs by two

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A_4)</td>
<td>(A_4)</td>
</tr>
<tr>
<td>(A_4 \leftarrow A_3);</td>
<td></td>
</tr>
<tr>
<td>repeat</td>
<td></td>
</tr>
<tr>
<td>3 Generate queue (Q_d);</td>
<td></td>
</tr>
<tr>
<td>for each (Q_d[i] \in Q_d) do</td>
<td></td>
</tr>
<tr>
<td>(A_4 \leftarrow A_4 \setminus Q_d[i]);</td>
<td></td>
</tr>
<tr>
<td>for each pair of locations (g_1, g_2 \in \Omega) do</td>
<td></td>
</tr>
<tr>
<td>Place two new APs at locations (g_1) and (g_2);</td>
<td></td>
</tr>
<tr>
<td>(A_4 \leftarrow A_4 \cup {\text{the two APs}});</td>
<td></td>
</tr>
<tr>
<td>Call Algorithm 2 to test the feasibility of (A_4);</td>
<td></td>
</tr>
<tr>
<td>if the output of algorithm 2 is FALSE then</td>
<td></td>
</tr>
<tr>
<td>(A_4 \leftarrow A_4 \cup Q_d[i]);</td>
<td></td>
</tr>
<tr>
<td>(A_4 \leftarrow A_4 \setminus {\text{the two APs}});</td>
<td></td>
</tr>
<tr>
<td>else</td>
<td></td>
</tr>
<tr>
<td>[ Return to the repeat statement;</td>
<td></td>
</tr>
<tr>
<td>until (i =</td>
<td>Q_d</td>
</tr>
<tr>
<td>Return (A_4);</td>
<td></td>
</tr>
</tbody>
</table>

Actually, we may continue to iteratively replace \(x\) nearby APs by \(x - 1\) \((x = 4, 5, \ldots)\), but there are two reasons that prevent us from doing that. The first one is about the computation time and time complexity for the next solution. The second one is about how much improvement we can obtain as compared to our last algorithm (see Section VI).

**B. Time Complexity Analysis**

The time complexity of our algorithm is determined by Algorithm 6, which calls Algorithm 2 to test the feasibility of the current AP placement scheme. In addition, Algorithm 2 calls Procedure 1 to obtain the throughput of STAs. Therefore, the time complexity of our algorithm is the product of the time complexity of Algorithm 6, Algorithm 2, and Procedure 1. Suppose that the input of Algorithm 6 is \(A\). In the worst case, we have to do \(|Q_d| = \binom{|A|}{3}\) attempts to replace three APs by two. In each replacement attempt, we should search \((\binom{|\Omega|}{2} + |\Omega|)\) times to find a feasible pair of locations for two new APs (if two new APs do not overlap, there are \(\binom{|\Omega|}{2}\) location combinations; if they overlap, there are \(|\Omega|\) candidate locations). Thus, the time complexity of Algorithm 6 is \(O(|A|^3 \binom{|\Omega|}{2} + |\Omega|)\). In Algorithm 2, for the network that contains \(|A|\) APs, when \(n\) APs fail, there
and power re-adjustment. Thus, the time complexity of Algorithm 2 is $O\left(\binom{|A|}{n}\right) = O\left(|A|^n\right)$. The time complexity of Procedure 1 is equal to the sum of the time complexity of its six steps, namely, $O\left(|S| + |A| + (|A|^2 + |A|) + |S| + |S| + |S|\right)$, where $(|A|^2 + |A|)$ is the time complexity of channel assignment and power re-adjustment. Thus, the time complexity of our algorithm is $O(|A|^3 + |\Omega|^2 (4|S| + |A|^2))$.

VI. Performance Evaluation

A. Simulation Settings

Since NS3 does not currently support some of the major functions (such as MU-OFDMA) of IEEE 802.11ax [32], we develop a simulator using MATLAB [33]. We set $C = \{1,2,\ldots,19\}$ as shown in Fig. 3 and $P = \{14,15,16,17\}$ dBm. For the path loss model [29], we set $G_{TX} = 4$ dBm, $G_{RX} = 4$ dBm, $P_{ref} = 30$ dBm, $\eta = 4$, and $\chi = 5$ dB. The frame decoding threshold and the interference signal strength threshold are set as: $\theta_B = -68$ dBm and $\theta_I = -77$ dBm. The RUs and the correspondence between the receiver minimum input level sensitivity and the data rate can be found in [3], which are applied to assign the RUs to STAs and obtain the data rate of STAs, i.e., $R_i$ ($i \in S$). For the throughput of STAs under the OFDMA mechanism (see Fig. 1), we set $L_{TF} = 68$ B, $L_{M,BA} = 118$ B, and $L_{OFDMA,BA} = 32$ B, where $L_{TF}$, $L_{M,BA}$, and $L_{OFDMA,BA}$ represent the size of TF, M-BA, and OFDMA-BA, respectively. When transmitting control frames (i.e., TF, M-BA, and OFDMA-BA), the data rate of APs/STAs is set as 7.5 Mbps [3]. The SIFS duration, $t_{SIFS}$, is set as $10 \times 10^{-6}$ s when operating in the 2.4 GHz band, and $16 \times 10^{-6}$ s when operating in the 5 GHz band [34]. We set the TXOP as $3 \times 10^{-3}$ s [35], which yields $t_{UL,PPDU} = 3 \times 10^{-3} - 2t_{SIFS} - t_{TF} - t_{M,BA}$. We suppose that the ratio of UL to DL duration is 1/2, which yields $t_{DL,PPDU} = 2t_{UL,PPDU}$. We assume that the lowest throughput that users can accept is 0.5 Mbps, that is, we set $\rho_L = 0.5$ Mbps. The following results are from an average of 30 simulation runs. In each run, the locations of STAs are generated randomly.

B. Effectiveness Evaluation

In this part, we set the target region as $50 \times 50$ m², which is divided into 25 $10 \times 10$ m² cells. Thus, $|\Omega| = 25$.

Considering that the throughput model in (4) plays a key role in our algorithm, we verify it first via the comparison of the simulation and the analytical results. Setting $|S| = 500$, $\beta = 90$, $n = 0$, and $\rho_H = 1$ Mbps, respectively, we obtain Tables III and IV, in which the data from row 2 to 6 represent the average number of APs and the average execution time of 30 runs, respectively. Table III shows that the number of APs obtained from the four-stage is equal to that obtained from the ES, namely, our four-stage algorithm performs as good as the ES under the small-scaled case. Table IV shows that the four-stage saves $41 \sim 82\%$ execution time as compared to ES. In short, our algorithm is very effective in providing a solution.

<table>
<thead>
<tr>
<th></th>
<th>$S$</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>400</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>ES method</td>
<td>2.00</td>
<td>3.00</td>
<td>3.00</td>
<td>3.20</td>
<td>4.00</td>
<td></td>
</tr>
<tr>
<td>Four-stage</td>
<td>2.00</td>
<td>3.00</td>
<td>3.00</td>
<td>3.20</td>
<td>4.00</td>
<td></td>
</tr>
<tr>
<td>Three-stage</td>
<td>2.10</td>
<td>3.00</td>
<td>3.73</td>
<td>3.83</td>
<td>5.07</td>
<td></td>
</tr>
<tr>
<td>Two-stage</td>
<td>2.57</td>
<td>3.07</td>
<td>4.13</td>
<td>5.40</td>
<td>7.53</td>
<td></td>
</tr>
<tr>
<td>One-stage</td>
<td>2.97</td>
<td>3.27</td>
<td>4.70</td>
<td>6.17</td>
<td>8.80</td>
<td></td>
</tr>
</tbody>
</table>

TABLE IV

<table>
<thead>
<tr>
<th></th>
<th>$S$</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>400</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>ES method</td>
<td>0.17</td>
<td>1.02</td>
<td>3.46</td>
<td>24.45</td>
<td>64.83</td>
<td></td>
</tr>
<tr>
<td>Four-stage</td>
<td>0.10</td>
<td>0.35</td>
<td>1.24</td>
<td>4.42</td>
<td>18.64</td>
<td></td>
</tr>
<tr>
<td>Three-stage</td>
<td>0.09</td>
<td>0.18</td>
<td>0.81</td>
<td>2.48</td>
<td>8.34</td>
<td></td>
</tr>
<tr>
<td>Two-stage</td>
<td>0.06</td>
<td>0.10</td>
<td>0.28</td>
<td>0.69</td>
<td>1.75</td>
<td></td>
</tr>
<tr>
<td>One-stage</td>
<td>0.04</td>
<td>0.06</td>
<td>0.14</td>
<td>0.28</td>
<td>0.59</td>
<td></td>
</tr>
</tbody>
</table>

Next, we compare our algorithm for individual subproblems against solutions from the literature. We replace our power adjustment method and channel assignment method in the four-stage algorithm by the method presented in [36] and [37], respectively, and compare them with the four-stage algorithm. Setting $|S| = 300, 400, 500, \beta = 90, n = 1, \rho_H = 1$ Mbps, respectively, we obtain Fig. 7, which illustrates that our power adjustment method and channel assignment method are better than existing ones.
C. Performance Evaluation

Because our problem is quite different from the others (see Table I), it is difficult to find suitable algorithms to compare with our algorithm. Thus, we evaluate the performance of our algorithm by comparing it with the Greedy algorithm and the Random methods under the larger area where we cannot find the optimal solution through an exhaustive search. We consider the stadium whose layout is shown in Fig. 8(a). The shaded area is the race area and the blank area has many seats for audience. The target region is the blank area which is divided into (100 × 80 − 60 × 40)/ (10 × 10) = 56 10 × 10 m² cells. To illustrate the advantages of our solution intuitively, an example is given in Fig. 8(b) ∼ (d), in which the red triangles, the blue circles, and the black circles represent the AP locations, the users, and the communication ranges of APs, respectively. The number next to the triangle is the number of APs being placed at that location. We observe that some locations are placed more than one AP. This is because some areas have denser users than the others. In addition, we allow APs to automatically adjust their power, so the communication ranges of them may be different from each other. Fig. 8 shows our solution is much more efficient than that of the Greedy algorithm and the Random method.

Next, we perform the performance comparison with various parameter settings. Setting |S| = 800, 900, 1000, β = 90, n = 1, and ρ_H = 1 Mbps, respectively, we obtain Fig. 9(a). From Fig. 9(a) we can see that the number of APs obtained from our algorithm is much smaller than that obtained from the Greedy algorithm and the Random method. Setting ρ_H = 1, 1.25, 1.5 (Mbps), |S| = 800, β = 90, and n = 1, respectively, we obtain Fig. 9(b), which shows that our algorithm produces the best result. Suppose that it costs 1 unit to deploy 1 AP, the cost savings of our algorithm are shown in Table V. We can see that our algorithm can save 32 ∼ 55 % in terms of deployment cost. In addition, we compare the execution time of the three methods in Table VI. The table shows that our algorithm has much higher time overhead than Greedy and random methods. The reason is that our algorithm gradually reduces the number of APs through four stages. We leave it as a future work to use parallel computing technique to reduce the execution time.

VII. CONCLUSION

In this paper, we deployed an IEEE 802.11ax-based dense WiFi network under the joint design of AP placement and power-channel-RU assignment. We formulated it as an optimization problem. We then analyzed the throughput of STAs under the mechanism of OFDMA and a widely used interference model and designed a heuristic algorithm with polynomial time complexity to solve the optimization problem which is NP-hard. Simulation results show that our algorithm is efficient and effective in reducing the number of APs.
REFERENCES


