Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 6929575, 14 pages
http://dx.doi.org/10.1155/2016/6929575

Research Article

Utility Maximization for Load Optimization in Cellular/WLAN Interworking Network Based on Generalized Benders Decomposition

Fanqin Zhou, Lei Feng, Peng Yu, Wenjing Li, and Luoming Meng

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Correspondence should be addressed to Wenjing Li; wjli@bupt.edu.cn

Received 21 July 2016; Accepted 3 October 2016

Academic Editor: Jung-Ryun Lee

Load steering is widely accepted as a key SON function in cellular/WLAN interworking network. To investigate load optimizing from a perspective of system utilization maximization more than just offloading to improve APs' usage, a utility maximization (UTMAX) optimization model and an ASRAO algorithm based on generalized Benders Decomposition are proposed in this paper. UTMAX is to maximize the sum of logarithmic utility functions of user data rate by jointly optimizing user association and resource allocation. To maintain the flexibility of resource allocation, a parameter β is added to the utility function, where smaller β means more resources can be allocated to edge users. As a result, it reflects a tradeoff between improvements in user throughput fairness and system total throughput. UTMAX turns out to be a mixed integer nonlinear programming, which is intractable intuitively. So ASRAO is proposed to solve it optimally and effectively, and an optional phase for expediting ASRAO is proposed by using relaxation and approximation techniques, which reduces nearly 10% iterations and time needed by normal ASRAO from simulation results. The results also show UTMAX's good effects on improving WLAN usage and edge user throughput.

1. Introduction

With the popularization of intelligent mobile terminals and enrichment of Internet services, it arouses surging demands for mobile Internet access in mobile users. So higher requirements on access capacity and data rate are put forward to commercially operated mobile networks. As spectrum band authorized to a mobile network is so scarce, academic and industrial research groups are continuously working on more spectrum-efficient radio communication technologies. Besides endeavors in this direction, taking advantages of available extra frequency bands seems much more economical and practical, which leads to a common deed among network operators of deploying WLAN in public areas, because WiFi works on the open ISM frequency band and wireless traffic can be offloaded from cellular network (CN) to WLAN.

A typical traffic offloading case in cellular/WLAN coexisted scenario is illustrated in Figure 1. In Figure 1(a), users (or UEs without distinction hereafter) are served dominantly by cellular base stations (BSs), while APs are underutilized. As service areas of CN and public WLAN overlap, to perform offloading, we can simply adjust user association and optimize resource allocation accordingly. The optimized scenario is expected to be as in Figure 1(b), where some users handover from BSs to nearby APs.

Initially, public access points (APs) in WLAN were deployed mainly in traffic hot-spots where many mobile users gathered, resulting in high usage of APs. However, with public WLAN gradually being a scale large enough to cover almost all urban areas, the problem is increasingly highlighted that most public WLAN APs are underutilized. The strict power limit of ISM devices is a root cause, while a lack of load distribution optimization approaches in present cellular/WLAN coexisting networks also plays a role.

So load steering between cellular and WLAN was an important research case in SEMAFOUR, a project in EU Framework Program 7 (EU-FP7) for developing multi-RAT/multilayer self-optimizing network (SON) function and integrated SON management system [1]. Its resulting approaches

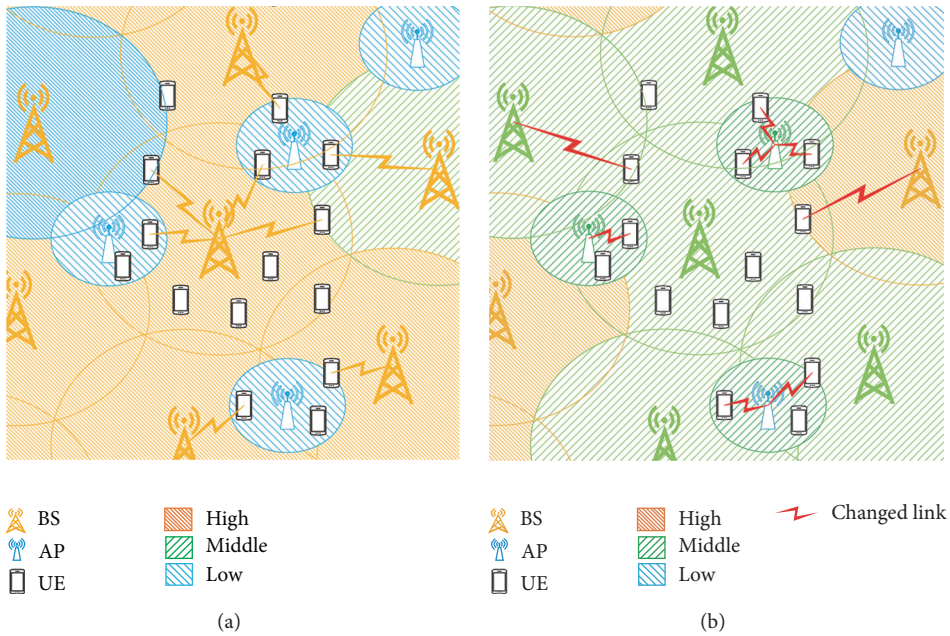

Figure 1: Offloading in cellular/WLAN interworking network.

pay more attention to applicability in practice, while a theoretical performance bound is not well investigated. A typical approach proposed is to make AP serve users as many as possible, which is sure to improve AP's usage to the most but will not produce the highest system utilization or overall users' satisfaction. Therefore, offloading is deemed as a basic purpose, but load optimizing is a further goal in cellular/WLAN interworking networks.

Moreover, providing better satisfactory data rate for users with the same infrastructure means more economical utilizing of investment, which is of more significance to commercially operated networks. To this end, optimizing user association and resource allocation to achieve ultimate overall users' satisfaction in cellular/WLAN interworking network are to be investigated in this paper.

Many prior works related to load optimizing in cellular/WLAN interworking networks have been surveyed in [2, 3]. In this paper, we roughly group the related approaches into three categories, according to whether limited, fair, or sufficient information exchange between CN and WLAN is required.

(1) UE-decision based approaches are widely used where information exchange is difficult, and statistical methods are used to estimate proper handover trigger criteria for individual UE on aspects, such as the average amount of data that could be transferred through CN or WLAN [4], the holding and continuing of an existing traffic data stream [5], the hysteresis for handover between CN and WLAN [6], the load status in a BS or AP [7], and the partial packet success [8]. These approaches focus on designing practical rules for individual UE to make proper handover decision without assists from networks.

(2) Network-assisted UE-decision based approaches become possible with more information attainable about a target network through information exchange, such as the achievable data rate in a target eNB or AP [9, 10], UE's velocity, and received signal strength of target AP [11]. These works use different methods, such as TOPSIS [10] and fuzzy logical system [11] to process multidimensional information obtained from network to help UE make better handover choice.

(3) Centralized optimization approaches are applicable when sufficient information is achievable. They usually focus on providing theoretical performance bounds for coexisting networks on various optimization goals, such as balancing resource utilization [12], minimizing power consumption [13, 14], maximizing quality of experience [15], and maximizing multilink aggregation [16]. These approaches may not be intended for application in actual networks, but they provide theoretical bounds for target optimization goals, which is significant for the design of practical network optimizing methods.

The existence of various types of approaches is partially due to the different requirements in actual practices of network optimizing and theoretical analyses in academic researches and largely due to the development of network technologies. In the early stage of cellular/WLAN coexisting network, the lack of information exchange between noninterworking CN and WLAN caused great difficulty in controlling user access or handover in a centralized way. Therefore, works at that time focused on UE-decision based approaches.

To improve the performance of coexisting networks, information exchange between networks is a prerequisite.

For this purpose, 3GPP proposed solutions for interworking between 3GPP network and WLAN in [17], and a media independent handover (MIH) standard was developed by IEEE in [18]. With more information attainable, especially that about load status in a target BS or AP (we use AN to denote a BS or AP hereafter), individual UE can even estimate its achievable data rate in a target BS or AP, so better handover decision can be made. On these bases, network-assisted UE-decision based became popular.

As multiple users may contend for access chances and resources, UE-decision based schemes neither can guarantee a resulting decision always to be the optimal nor can tell where a performance bound goes from a whole network perspective. Therefore, centralized optimization approaches are widely investigated to derive such a theoretical performance bound. These approaches are usually of high computational complexity; however, deriving such a performance bound will give guidance and direction for more practical approaches with better tradeoff between optimality and complexity.

Therefore, this paper is to derive a centralized logarithmic utility maximization model to improve overall users' satisfaction by jointly optimizing user association and resource allocation. The key contributions are summarized as follows.

(i) A utility maximization (UTMAX) optimization model is formulated to depict the users' satisfaction degree maximization by jointly planning association selection and resource allocation (ASRA). Instead of directly using a logarithmic function of data rate to model user's satisfaction of a data rate, which produces identical resource allocation in a same BS and thus lack flexibly in resource allocation, we add a control parameter β to it. With smaller β value, the users with poor channel conditions are to be assigned more resources and vice versa. Therefore, a tradeoff between user throughput fairness and system throughput gain can be achieved by selecting proper β value.

(ii) The derived UTMAX has both binary and continuous variables for user association and resource allocation, respectively, which is a mixed integer nonlinear programming (MINLP) and seems intractable intuitively. To solve it optimally and effectively, we devise an ASRAO algorithm on the basis of Bender Decomposition (BD) framework [19, 20]. We also propose optional procedures for ASRAO to expedite the computation of UTMAX using relaxation and approximation techniques [21], which dramatically reduces the number of iterations needed before convergence, especially when a UTMAX instance is in large scale. In addition, the convergence of our algorithms is validated both theoretically and experimentally in this paper.

(iii) Our work contributes to SON. The UTMAX and its corresponding ASRAO can serve directly in the scheme-planning module, which forms a self-optimizing closed loop together with monitor, evaluation, and execution modules, to generate the optimal ASRA scheme through a centralized way. It also provides a performance upper bound for load optimization on purposes of a system preferred tradeoff between overall throughput and end user satisfaction. So It can guide the development of novel suboptimal but less computationally complex algorithms by evaluating the performance gaps between them.

The rest of the paper are arranged as follows. In Section 2 we formulate the user data rate and propose the UTMAX optimization model. Section 3 analyzes a special property of UTMAX, and, to exploit this property, we propose the ASRAO algorithm based on generalized Benders Decomposition (GBD) to solve it. Optional procedures for ASRAO to expedite the computing of UTMAX are also given in this section. In Section 4, we evaluate the performance of (A-)ASRAO and effect of UTMAX through extensive simulations. The paper will be concluded in Section 5.

2. System Model and Problem Formulation

A cellular/WLAN interworking network is composed of J BSs and K APs, serving I UEs. We denote by $\mathcal{J}$, $\mathcal{K}$, and $\mathcal{I}$ the sets of all BSs, APs, and UEs, respectively. UE is served uniquely by a BS or AP, from which it gets largest received power. The association between UE i and AN l is indicated by binary variable x_{il}, which equals 1 if UE i is served by AN or 0 otherwise as in (1). A matrix $\mathbf{X} = (x_{il})_{i\in\mathcal{I},l\in\mathcal{J}\cup\mathcal{K}}$ represents associations between all UEs and ANs. UE is uniquely served by one AN, namely, a BS or AP, which forms the constraint in (2).

$$x_{il} = \begin{cases} 1, & \text{UE } i \text{ is served by AN } l, \\ 0, & \text{otherwise,} \end{cases} \tag{1}$$

$$\sum_{l\in\mathcal{J}\cup\mathcal{K}} x_{il} = 1. \tag{2}$$

2.1. Channel and Resource Model of CN. The access rate of a UE in CN is determined by two factors, the channel condition and the allocated resource. $\mathbf{\Lambda}_{\mathcal{J}} = (\lambda_{ij})$, $i \in \mathcal{I}$, $j \in \mathcal{J}$ represents the resource allocation scheme in CN, in which λ_{ij} denotes the ratio of resource allocated to UE i from BS j. To assure a basic quality of service (QoS), a minimal ratio of resource, λ_ε, should be allocated to UE i from BS j if $x_{ij} = 1$, and $\lambda_{ij} = 0$ if $x_{ij} = 0$, which can be summarized in (3). The channel condition between UE i and BS j can be indicated by SINR_{ij}, as in (4), in which Pr_{ij} is the received signal power of UE i from BS j, and α is a coefficient of cochannel interference. According to Shannon Formula, the data rate of UE i served by BS j, r_{ij} can be expressed in (5), wherein $R_{ij} = W_j \log(1 + \mathrm{SINR}_{ij})$.

$$\lambda_\varepsilon x_{ij} \le \lambda_{ij} \le x_{ij}, \tag{3}$$

$$\mathrm{SINR}_{ij} = \frac{\mathrm{Pr}_{ij}}{\alpha \cdot \sum_{i'\in B/\{i\}} \mathrm{Pr}_{i'j} + \sigma^2}, \tag{4}$$

$$r_{ij} = \lambda_{ij} W_j \log_2\left(1 + \mathrm{SINR}_{ij}\right) = \lambda_{ij} W_j \gamma_{ij} = \lambda_{ij} R_{ij}. \tag{5}$$

2.2. Channel and Resource Model of WLAN. Due to the sparse deployment of public WLAN APs and the limited transmit power, for simplification, the interferences between WLAN APs will not be taken into consideration. Reference [9] gives the way to evaluate the service rate of UE $i \in \mathcal{I}$ accessing public WLAN AP $k \in \mathcal{K}$, as in

$$r_{ik} = C_{\text{WLAN}} \cdot \frac{\text{MCS}_{\text{WLAN}}\left(\text{Pr}_{ik}\right)}{\max\left\{\text{MCS}_{\text{WLAN}}\left(\cdot\right)\right\}} \quad i \in \mathcal{I},\ k \in \mathcal{K}, \quad (6)$$

wherein C_{WLAN} denotes the average data rate evaluated by AP when the modulation and coding scheme (MCS) with the highest spectrum efficiency is used and Pr_{ik} is the received power strength of UE i from WLAN AP k. $\text{MCS}_{\text{WLAN}}(\cdot)$ is corresponding to each possible Pr_{ik} in MCS table. Taking resource allocation into consideration, we have $C_{\text{WLAN}} = C_{\text{WLAN\_FULL}}\lambda_{ik}$, where $C_{\text{WLAN\_FULL}}$ is the available data rate if the observing UE occupies resources exclusively in AP. As interference to AP is not taken into consideration and only a limited number of UEs are served in each AP, we assume $C_{\text{WLAN\_FULL}}/\max\{\text{MCS}_{\text{WLAN}}(\cdot)\} = 1$. Therefore, $r_{ik} = \lambda_{ik}R_{ik}$.

AP uses Round Robin (RR) for packet data scheduling, which from long time average perspective can be viewed as UEs in AP equally sharing channel resources (7). Network Operators commonly limit the data rate of UEs in WLAN APs to avoid resources being occupied dominantly by user requesting big volume of data with good channel condition. For simplification, in this paper we assume the ratio of resources occupied by UE in AP has an upper bound $\lambda_m \leq 1$ as in (8). In addition, UEs contend to access AP via shared wireless channel, and too many UEs would severely deteriorate the throughput performance, so the number of UEs served simultaneously by AP has an upper bound N_m as in (10), which amounts to setting a lower bound to resources allocated to UE. Hence, we have the following constraints on resource allocation in AP.

$$\lambda_{ik} = \lambda_k^{\text{avr}} x_{ik}, \quad i \in \mathcal{I}_k,\ k \in \mathcal{K}, \quad (7)$$

$$\lambda_{ik} \leq \lambda_m x_{ik}, \quad (8)$$

$$\sum_{i \in \mathcal{I}} \lambda_{ik} \leq 1, \quad (9)$$

$$\sum_{i \in \mathcal{I}} x_{ik} \leq N_m. \quad (10)$$

As UE is served uniquely by a BS or AP, the overall data rate of UE i can be formulated as $r_i = \sum_i \lambda_{il} R_{il}$. It should be noted that R_{il} is not a variable to be optimized in our model that means more accurate approaches for full band rate estimation can be used, which is not a key point in this paper and we will pay no more attention to it. After formulating the data rate of UE in CN and WLAN, we take a utility function perspective to investigate a utility maximization problem for the data rate r_i of each UE i in the system, to get the optimal association and resource allocation.

2.3. Utility Maximization Problem. For utility maximization, utility function $U_i(\cdot)$ is often chosen to be continuously differentiable, monotonically increasing, and strictly concave. UE i gets utility $U_i(r_i)$ when its receiving rate is r_i. The overall system utility can then be represented as $\sum_i U_i(r_i)$. Maximize it with some practical constraints, we can get the general utility maximization problem. $U_i(\cdot)$ should be chosen carefully. If $U_i(\cdot)$ is a linear function, $\max \sum_i U_i(r_i)$ turns out to be a throughput maximization and it produces a trivial solution, where each BS allocates resources only to its user with best channel quality. It is not a satisfactory solution for a multiple user system. So we seek a utility function that would naturally encourage fairness resource allocation among users. To this end, a logarithmic function is widely used. It is concave and has diminishing returns. This property approximates the fact that a well-served user has low priority in getting more resources, so it encourages load balancing and improves overall users' satisfaction.

However, logarithmic utility function, like in [15], always results in equal resource allocation among users in a same BS. Because $\sum_i \log(R_i\lambda_i) = \sum_i \log(R_i) + \sum_i \log(\lambda_i)$, the utility of individual BS has nothing to do with R_i, and $\sum_i \log(\lambda_i) = \log(\prod_i \lambda_i) \leq n \log(\sum_i \lambda_i / n)$ always hold. It reaches maximum only when resources are all shared equally, $\lambda_{i_1} = \lambda_{i_2}$, $\forall i_1, i_2 \in \mathcal{I}_l$. This lacks flexibility in resource allocation, so we slightly adjust this logarithm utility function and add a BS related parameter β_l to r_i, to make resource allocation scheme represent certain system preference, like caring more about fairness or demanding higher system throughput. If $b_l = 0$, it means an identical resource allocation. If $b_l > 0$, users with better channel condition get more resources and vice versa. The logarithmic utility function then becomes $\log(\sum_l R_{il}\lambda_{il} + \beta_l x_{il})$. As x_{il} is binary variable and $\sum_l x_{il} = 1$, $\log(\sum_l R_{il}\lambda_{il} + \beta_l x_{il})$ can be transformed into $\sum_l x_{il} \log(\sum_l R_{il}\lambda_{il} + \beta_l)$. And our utility maximization problem is formed as P1 in P1, where for simplicity we assume in CN $\lambda_m = 1$, and in WLAN $\lambda_\varepsilon = 0$.

$$\begin{aligned}
\text{P1:}\ \max_{(\mathbf{\Lambda},\mathbf{X})}\quad & \Phi\left(\mathbf{\Lambda},\mathbf{X}\right), \\
\text{s.t.}\quad & \Phi\left(\mathbf{\Lambda},\mathbf{X}\right) \\
& = \left(\sum_i \sum_l \left(x_{il} \log\left(\sum_l R_{il}\lambda_{il} + \beta_l\right)\right)\right), \\
& \sum_l x_{il} = 1, \\
& \sum_i \lambda_{il} \leq 1, \\
& \lambda_\varepsilon x_{il} \leq \lambda_{il} \leq \lambda_m x_{il}, \\
& \lambda_{ik} = \lambda_k^{\text{avr}} x_{ik}, \\
& \sum_{i \in \mathcal{I}} x_{ik} \leq N_m, \\
& x_{il} \in \{0,1\}, \\
& i \in \mathcal{I}, \\
& k \in \mathcal{K}, \\
& l \in \mathcal{J} \cup K.
\end{aligned} \quad (11)$$

P1 has both binary and continuous variables and nonlinear part in goal function. These properties make it an MINLP, which is intractable intuitively. However, this problem has a special structure, utilizing which we propose ASRAO algorithm based on GBD to solve it effectively. The problem analyses and ASRAO algorithm will be presented in next section.

3. ASRAO Algorithm

3.1. Problem Analyses. Assume association indicator $\mathbf{X}$ is fixed, then each UE i is served by a specific AN l, and each AN l has a fixed UE collection $\mathcal{I}_l = \{i \mid x_{il} = 1\}$. UE i contributes $\log(R_{il}\lambda_{il} + \beta_l)$ to utility of AN l. Its value is determined by λ_{il}, which is only related to AN l. So resource allocation in each AN can be solved independently with given user association $\dot{\mathbf{X}}$. Therefore, $\Phi(\boldsymbol{\Lambda}, \dot{\mathbf{X}})$ can be identically transformed to

$$\begin{aligned}\Phi\left(\boldsymbol{\Lambda}, \dot{\mathbf{X}}\right) &= \sum_{\substack{i\in\mathcal{I},\\ l\equiv\arg\{x_{il}=1\}}} \log\left(R_{il}\lambda_{il} + \beta_l\right) \\ &= \sum_{l\in\mathcal{I}\cup K}\left(\sum_{i\in I_l}\log\left(R_{il}\lambda_{il} + \beta_l\right)\right) \\ &= \sum_{l\in\mathcal{I}\cup K}\Phi_l\left(\boldsymbol{\lambda}_l, \dot{\mathbf{x}}_l\right),\end{aligned} \tag{12}$$

where $\Phi_l(\boldsymbol{\lambda}_l, \dot{\mathbf{x}}_l) = \sum_{i\in I_l}\log(R_{il}\lambda_{il} + \beta_l)$ is the utility of AN l. So we get a subproblem of P1 on condition of $\dot{\mathbf{X}}$.

$$\begin{aligned}\max_{(\boldsymbol{\Lambda})}\quad & \sum_{i\in I_l}\log\left(R_{il}\lambda_{il} + b_l\right) \\ \text{s.t.}\quad & \sum_i \lambda_{il} \le 1, \\ & \lambda_\varepsilon \dot{x}_{il} \le \lambda_{il} \le \lambda_m \dot{x}_{il}, \\ & \lambda_{ik} = \lambda_k^{\text{avr}} \dot{x}_{ik}.\end{aligned} \tag{13}$$

This is a classical problem in information theory and a water filling (WF) principle is specially developed to solve it, which is derived from Karush-Kuhn-Tucker (KKT) optimal conditions. It provides direction solutions to problems with the form like (13). To enable its applicability in P1, we extend WF to a multi-AN case and denote its results as MWF, short for multi-AN water filling. With $\mathbf{X}$ fixed as $\dot{\mathbf{X}}$ in P1, we have a MWF applicable problem as in

$$\begin{aligned}\max_{(\boldsymbol{\Lambda})}\quad & \Phi\left(\boldsymbol{\Lambda}, \dot{\mathbf{X}}\right), \\ \text{s.t.}\quad & \Phi\left(\boldsymbol{\Lambda}, \dot{\mathbf{X}}\right) \\ & = \left(\sum_i\sum_l\left(\dot{x}_{il}\log\left(\sum_l R_{il}\lambda_{il} + \beta_l\right)\right)\right), \\ & \lambda_\varepsilon \dot{x}_{il} \le \lambda_{il} \le \lambda_m \dot{x}_{il}, \\ & \sum_i \lambda_{il} \le 1, \\ & \lambda_{ik} = \lambda_k^{\text{avr}} x_{ik}.\end{aligned} \tag{14}$$

According to generalized duality theory, the dual function of $\Phi(\boldsymbol{\Lambda}, \dot{\mathbf{X}})$ is as in

$$\begin{aligned}&\mathscr{L}\left(\boldsymbol{\Lambda}, \boldsymbol{\nu}, \boldsymbol{\omega}, \boldsymbol{\gamma}, \dot{\mathbf{X}}\right) \\ &= \left(\sum_i\sum_l\left(\dot{x}_{il}\log\left(\sum_l R_{il}\lambda_{il} + \beta_l\right)\right)\right) \\ &\quad - \sum_l \nu_l\left(\sum_i \lambda_{il} - 1\right) - \sum_i\sum_l \omega_{il}\left(\lambda_{il} - \dot{x}_{il}\right) \\ &\quad + \sum_i\sum_l \gamma_{il}\left(\lambda_{il} - \varepsilon\dot{x}_{il}\right) - \sum_i\sum_k \mu_{ik}\left(\lambda_{ik} - \dot{x}_{ik}\lambda_k^{\text{avr}}\right),\end{aligned} \tag{15}$$

where $\boldsymbol{\nu}$, $\boldsymbol{\omega}$, $\boldsymbol{\gamma}$ are nonnegative relaxed variables and $\boldsymbol{\mu}$ is nonzero. The local optimal is reached, when the following KKT conditions are satisfied.

$$\begin{aligned}\frac{\partial\Phi(\boldsymbol{\Lambda})}{\partial\lambda_{ij}} &= \frac{R_{ij}}{\sum_j R_{ij}\lambda_{ij} + \sum_l \dot{x}_{ij}b_j} - \nu_j - \omega_{ij} \\ &\quad + \gamma_{ij} = 0, \\ \frac{\partial\Phi(\boldsymbol{\Lambda})}{\partial\lambda_{ik}} &= \frac{R_{ik}}{\sum_k R_{ik}\lambda_{ik} + \sum_l \dot{x}_{ik}b_k} - \nu_k - \omega_{ik} \\ &\quad + \gamma_{ik} - \mu_{ik} = 0, \\ \frac{\partial\Phi(\boldsymbol{\Lambda})}{\partial\lambda_k^{\text{avr}}} &= \mu_{ik}\dot{x}_{ik} = 0, \\ \omega_{il}\left(\lambda_{il} - \dot{x}_{il}\right) &= 0, \\ \nu_l\left(\sum_i \lambda_{il} - 1\right) &= 0, \\ \gamma_{il}\left(\lambda_{il} - \varepsilon\dot{x}_{il}\right) &= 0, \\ \mu_{ik}\left(\lambda_{ik} - \dot{x}_{ik}\lambda_k^{\text{avr}}\right) &= 0, \\ \nu_l, \omega_{il}, \gamma_{il} &\in [0, +\infty), \ \mu_{ik} \ne 0.\end{aligned} \tag{16}$$

The solutions to (16) are as follows. MWF results for CN is as in

$$\begin{aligned}\nu_j^* &= \left(\sum_i \dot{\bar{x}}_{ij}\right)\left(\overline{\rho} + \sum_i \frac{b_j\dot{\bar{x}}_{ij}}{R_{ij}}\right)^{-1}, \\ \lambda_{ij}^* &= \max\left\{\dot{x}_{ij}\left(\frac{1}{\nu_j^*} - \frac{\beta_j}{R_{ij}}\right), 0\right\} + \lambda_\varepsilon\dot{x}_{ij}, \\ \gamma_{ij} &= \begin{cases} -\left(\dfrac{R_{ij}}{R_{ij'}\lambda_{ij'} + \beta_{j'}} - \nu_j^*\right), & \dfrac{R_{ij}}{R_{ij'}\lambda_{ij'} + \beta_{j'}} \le \nu_j^*, \\ 0, & \text{else}, \end{cases} \\ \omega_{ij} &= \left(\frac{R_{ij}}{R_{ij'}\lambda_{ij'} + \beta_{j'}} - \nu_j^*\right) + \gamma_{ij}.\end{aligned} \tag{17}$$

The following (18) is MWF results for WLAN.

$$\lambda_k^{\text{avr}} = \begin{cases} \min\left\{\lambda_m, \dfrac{1}{\sum_i \dot{x}_{ik}}\right\}, & \sum_i \dot{x}_{ik} > 0, \\ 0 & \sum_i \dot{x}_{ik} = 0, \end{cases}$$

$$\lambda_{ik}^* = \lambda_k^{\text{avr}} \dot{x}_{ik},$$

$$\nu_k^* = \begin{cases} \min\left\{\dfrac{R_{ik}}{\beta_k + \sum_i \dot{x}_{ik}}\right\}, & \lambda_m \sum_i \dot{x}_{ik} \geq 1, \\ 0, & \lambda_m \sum_i \dot{x}_{ik} < 1, \end{cases}$$

$$\mu_{ik} = \begin{cases} -\left(\dfrac{R_{ik}}{R_{ik'}\lambda_{ik'} + \beta_{k'}} - \nu_k^*\right), & \lambda_m \sum_i \dot{x}_{ik} > 1, \\ 1, & \lambda_m \sum_i \dot{x}_{ik} \leq 1, \end{cases} \tag{18}$$

$$\gamma_{ik} = \begin{cases} -\left(\dfrac{R_{ik}}{R_{ik'}\lambda_{ik'} + \beta_{k'}} - \nu_k^*\right), & \dfrac{R_{ik}}{R_{ik'}\lambda_{ik'} + \beta_{k'}} < \nu_k^*, \\ 0, & \text{else}, \end{cases}$$

$$\omega_{ik} = \left(\frac{R_{ik}}{R_{ik'}\lambda_{ik'} + \beta_{k'}} - \nu_k^*\right) + \gamma_{ik} + \mu_{ik}.$$

Using above MWF results, we can directly calculate optimal $\mathbf{\Lambda}$ with a given $\dot{\mathbf{X}}$. To utilize MWF in solving problem P1, an optimal $\mathbf{X}$ should be retrieved. However, as $\mathbf{X}$ is a matrix of binary variable entries and $\mathbf{\Lambda}$ is closely related with $\mathbf{X}$, it is difficult to find such an optimal. In this paper, instead of directly solving an optimal $\mathbf{X}$, we devise ASRAO algorithm to solve it iteratively and effectively. ASRAO is originated from BD, which provides a framework for addressing mixed integer programming problems by decomposing it into two smaller subproblems and solving them iteratively. The details are in the sequel.

3.2. Generalized Benders Decomposition. BD is originally proposed to solve mixed integer linear programming (MILP). Instead of solving all variables and constraints simultaneously, it decomposes an MILP into two subproblems, a master problem (MP) and a slave problem (SP), and the original MILP can be solved by solving MP and SP iteratively. In BD frameworks, MP is an MILP problem which consists of all integer variables in the original problem, while SP is a Linear Programming problem with all the continuous variables from the original problem. In each iteration, BD utilizes an extreme point or extreme ray derived from the dual of SP to generate an optimality cut or feasibility cut to trim the feasible domain of MP, and MP will eventually reach the same optima as the original MILP.

GBD extends the BD approach to a more general class of problems by adopting nonlinear duality theory, and some nonlinear problems are brought into range. When solving MINLP, GBD follows the framework of BD, but SP is derived as NLP. To generate the feasibility cuts and optimality cuts, GBD utilized KKT conditions to calculate an extreme point or extreme ray of the Lagrange dual functions of NLP SP. A GBD applicable problem has a general form as in

$$\begin{aligned} \max_{(\mathbf{x},\mathbf{y})} \quad & F(\mathbf{x}, \mathbf{y}), \\ \text{s.t.} \quad & G(\mathbf{x}, \mathbf{y}) \geq 0, \\ & H(\mathbf{x}, \mathbf{y}) = 0, \\ & \mathbf{x} \in \{0,1\}_{m\times 1}, \\ & \mathbf{y} \in D \subseteq \mathbb{R}_+^{n\times 1}. \end{aligned} \tag{19}$$

It should meet following situations: (a) for fixed $\mathbf{x}$, (19) separates into a number of independent optimization problems; (b) for fixed $\mathbf{x}$, (19) assumes a well-known special structure that efficient solution procedures are available; (c) (19) may be not concave program in $\mathbf{x}$ and $\mathbf{y}$ jointly, but fixing $\mathbf{x}$ renders it so in $\mathbf{y}$. According to the analyses, UTMAX obviously satisfies these conditions.

The initial MP of (19) in GBD procedures is as follows:

$$\begin{aligned} \max_{(\mathbf{x},\eta)} \quad & \eta, \\ \text{s.t.} \quad & G(\mathbf{x}) \geq 0, \\ & H(\mathbf{x}) = 0, \\ & \eta \in \mathbb{R}, \\ & \mathbf{x} \in E \subseteq \{0,1\}_{m\times 1}, \end{aligned} \tag{20}$$

where $G(\mathbf{x}) \geq 0$ and $H(\mathbf{x}) = 0$ are constraints from (19) that only related with $\mathbf{x}$. Solving a solution $(\mathbf{x}^{(t)}, \eta)$ from MP, SP can be derived by simply fixing $\mathbf{x}$ in (19) as $\mathbf{x}^{(t)}$,

$$\begin{aligned} \max_{(\mathbf{y})} \quad & F(\mathbf{x}^{(t)}, \mathbf{y}), \\ \text{s.t.} \quad & G(\mathbf{x}^{(t)}, \mathbf{y}) \geq 0, \\ & H(\mathbf{x}^{(t)}, \mathbf{y}) = 0, \\ & \mathbf{y} \in D \subseteq \mathbb{R}_+^{n\times 1}, \end{aligned} \tag{21}$$

where $\mathbf{x}^{(t)}$ is the solution to MP in tth iteration. The Lagrange function to problem (14) is $\mathscr{L}(\mathbf{y}, \nu, \upsilon) = F(\mathbf{x}^{(t)}, \mathbf{y}) + \nu G(\mathbf{x}^{(t)}, \mathbf{y}) + \upsilon H(\mathbf{x}^{(t)}, \mathbf{y})$. If optimal solution $\mathbf{y}^*$ to SP exists, it can be derived by solving KKT point $(\mathbf{y}^{(u)}, \nu^{(u)}, \mu^{(u)})$, and due to the strictly convex property of SP, $\mathscr{L}(\mathbf{y}^{(u)}, \nu^{(u)}, \mu^{(u)}) = \max_{(\mathbf{y})} F(\mathbf{x}^{(t)}, \mathbf{y})$. After getting $(\mathbf{y}^{(u)}, \nu^{(u)}, \mu^{(u)})$, an optimality cut $\Gamma(\mathbf{x}, \mathbf{y}^{(u)}, \nu^{(u)}, \upsilon^{(u)}) \geq \eta$ can be generated and added to MP, which actually constructs an upper bound for MP. If the optimal solution to SP does not exist, a feasibility cut should be added to cut off the infeasible $\mathbf{x}^{(v)} = \mathbf{x}^{(t)}$. Specifically, the feasible cut is $\sum_{i\in Y_v} x_i - \sum_{i\in N_v} x_i \leq |Y_v| - 1$, derived from $\mathbf{x} \neq \mathbf{x}^{(v)}$, wherein Y_v is the set of indexes indicating the nonzero elements in $\mathbf{x}^{(v)}$ and N_v is that indicating the zero ones.

Therefore, the MP in tth iteration is as

$$\begin{aligned}
\max_{(\mathbf{x},\eta)} \quad & \eta, \\
\text{s.t.} \quad & \Gamma\left(\mathbf{x}, \mathbf{y}^{(u)}, \nu^{(u)}, \upsilon^{(u)}\right) \geq \eta, \quad \forall u = 1, 2, \ldots, t_1, \\
& \sum_{(i,l)\in Y_v} - \sum_{(i,l)\in N_v} \leq \left|Y_v\right| - 1, \quad \forall v = 1, 2, \ldots, t_2, \\
& G(x) \geq 0, \\
& H(x) = 0, \\
& \mathbf{x} \in (0,1)_{m\times 1},
\end{aligned} \tag{22}$$

where $t_1 + t_2 = t$. Based on above problem decomposition framework, we propose the following ASRAO algorithm to solve P1.

3.3. ASRAO Algorithm. As is depicted in BD framework, problem P1 can be decomposed into MP and SP as in (26) and (24), separately. The initial MP to be solved is

$$\begin{aligned}
\max_{(\mathbf{X})} \quad & \eta, \\
\text{s.t.} \quad & \sum_{l\in\mathcal{F}\cup\mathcal{K}} x_{il} = 1, \\
& \sum_{i\in\mathcal{F}} x_{ik} \leq N_m, \\
& x_{il} \in \{0,1\}, \\
& i \in \mathcal{F}, \\
& l \in \mathcal{J} \cup \mathcal{K}.
\end{aligned} \tag{23}$$

In the tth iteration, denote the solution to MP as $(\mathbf{X}^{(t)}, \eta^{(t)})$, and the SP has the form

$$\begin{aligned}
\max_{(\mathbf{\Lambda})} \quad & \Phi\left(\mathbf{\Lambda}, \mathbf{X}^{(t)}\right), \\
\text{s.t.} \quad & \left(\mathbf{\Lambda}, \mathbf{X}^{(t)}\right) \\
& = \left(\sum_i \sum_l \left(x_{il}^{(t)} \log\left(\sum_l R_{il}\lambda_{il} + \beta_l\right)\right)\right), \\
& \lambda_\varepsilon x_{il}^{(t)} \leq \lambda_{il} \leq \lambda_m x_{il}^{(t)}, \\
& \lambda_{ik} = \lambda_{ik}^{\text{avr}} x_{ik}, \\
& \sum_{i\in\mathcal{F}} \lambda_{ij} \leq 1, \quad \lambda_{ij} \in \mathbb{R}_+.
\end{aligned} \tag{24}$$

If optimal solution exists in problem (24), the KKT point is $(\mathbf{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}) = (\mathbf{\Lambda}_{\mathcal{F}}^{(t)}, \boldsymbol{\nu}^{(t)}, \boldsymbol{\omega}^{(t)}, \boldsymbol{\gamma}^{(t)}, \boldsymbol{\mu}^{(t)})$, and an optimal cut $\Gamma(\mathbf{X}, \mathbf{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}) \geq \eta$ should be added to MP, wherein

$$\begin{aligned}
& \Gamma\left(\mathbf{X}, \mathbf{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}\right) \\
& = \left(\sum_i \sum_l \left(x_{il} \log\left(\sum_l R_{il}\lambda_{il}^{(u)} + \beta_l\right)\right)\right) \\
& \quad - \sum_l \nu_l^{(u)} \left(\sum_i \lambda_{il}^{(u)} - 1\right) \\
& \quad - \sum_i \sum_l \omega_{il}^{(u)} \left(\lambda_{il}^{(u)} - \lambda_m x_{il}\right) \\
& \quad + \sum_i \sum_l \gamma_{il}^{(u)} \left(\lambda_{il}^{(u)} - \lambda_\varepsilon x_{il}\right) \\
& \quad - \sum_i \sum_k^{(u)} \mu_{ik}^{(u)} \left(\lambda_{ik}^{(u)} - x_{ik}\lambda_k^{\text{avr}(u)}\right).
\end{aligned} \tag{25}$$

Due to strong duality of SP in (24), we have $\Phi(\mathbf{\Lambda}^{(u)}, \mathbf{X}^{(t)}) = \Gamma(\mathbf{X}^{(t)}, \mathbf{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)})$. If an optimal solution does not exist, a feasibility cut $\sum_{(i,l)\in Y_v} x_{il} - \sum_{(i,l)\in N_v} x_{il} \leq |Y_v| - 1$ is added to MP to cut off the infeasible $\mathbf{X}^{(v)} = \mathbf{X}^{(t)}$, where $Y_v = \{(i,l) \mid x_{il} = 1,\ i \in \mathcal{F},\ l \in \mathcal{J} \cup \mathcal{K}\}$ and $N_v = \{(i,l) \mid x_{il} = 0,\ i \in \mathcal{F},\ l \in \mathcal{J} \cup \mathcal{K}\}$. And the derived MP in tth iteration is as (26) and $t_1 + t_2 = t$.

$$\begin{aligned}
\max_{(\mathbf{X}_{\mathcal{J}}, \mathbf{X}_{\mathcal{K}}, \eta)} \quad & \eta, \\
\text{s.t.} \quad & \Gamma\left(\mathbf{X}, \mathbf{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}\right) \geq \eta, \\
& \qquad\qquad \forall u = 1, \ldots, t_1, \\
& \sum_{(i,l)\in Y_v} x_{il} - \sum_{(i,l)\in N_v} x_{il} \leq \left|Y_v\right| - 1, \\
& \qquad\qquad \forall v = 1, \ldots, t_2, \\
& \sum_{l\in\mathcal{F}\cup\mathcal{J}} x_{il} = 1, \\
& \sum_{i\in\mathcal{F}} x_{ik} \leq N_m, \\
& x_{il} \in \{0,1\}, \\
& i \in \mathcal{F}, \\
& l \in \mathcal{F} \cup \mathcal{J}.
\end{aligned} \tag{26}$$

Denote the lower bound and upper bound of problem P1 as $LB^{(t)}$ and $UB^{(t)}$, respectively. They can be derived from problem (24) and (26), which will be proved later in Lemma 1. The iteration procedure terminates when the gap between $UB^{(t)}$ and $LB^{(t)}$ becomes zero.

The pseudocodes of the above procedures are as shown in Algorithm 1.

(1) **Initialize**: $UB^{(0)} = +\infty$, $LB^{(0)} = -\infty$, Let $t = 0$, $u = 0$, $v = 0$.
(2) **repeat**
(3) Let $t = t + 1$ and solve (26) to obtain current optimal solution $(\mathbf{X}^{(t)}, \boldsymbol{\eta}^{(t)})$, $UB^{(t)} = \eta^{(t)}$
(4) **if** (24) with $\mathbf{X}^{(t)}$ is bounded: $u = u + 1$, and solve it with MWF to get a KKT point $(\boldsymbol{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}) = (\boldsymbol{\Lambda}^{(t)}, \boldsymbol{\nu}^{(t)}, \boldsymbol{\omega}^{(t)}, \boldsymbol{\gamma}^{(t)}, \boldsymbol{\mu}^{(t)})$. The lower bound is set to $LB^{(t)} = \max\{\max_{1\le s\le t}\{\Phi(\boldsymbol{\Lambda}^{(s)}, \mathbf{X}^{(s)})\}, LB^{(0)}\}$ and add $\Gamma(\mathbf{X}, \boldsymbol{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}) \ge \eta$ to (26)
(5) **elseif** (24) with $\mathbf{X}^{(t)}$ is infeasible: $v = v + 1$, $\mathbf{X}^{(v)} = \mathbf{X}^{(t)}$ $UB^{(t)} = UB^{(t-1)}$, and add $\sum_{(i,l)\in Y_v} x_{il} - \sum_{(i,l)\in N_v} x_{il} \le |Y_v| - 1$ to (24).
(6) **else** as (24) is unbounded, P1 is unbounded, ASRAO stops and return an indication of infeasible problem.
(7) **until** $UB^{(t)} - LB^{(t)} \le 0$
(8) Return $(\mathbf{X}^*, \boldsymbol{\Lambda}^*) = (\mathbf{X}^{(t)}, \boldsymbol{\Lambda}^{(t)})$ as the optimal solution to problem P1.

Algorithm 1: The ASRAO algorithm.

Lemma 1. *In tth iteration, denote the values of goal functions in problem (24) and (26) as $L^{(t)}$ and $U^{(t)}$, and then $LB^{(t)} = \max_{0\le s\le t}\{L^{(t)}\} = \max_{0\le s\le t}\{\Phi(\boldsymbol{\Lambda}^{(s)}, \mathbf{X}^{(s)})\}$ and $UB^{(t)} = L^{(t)} = \eta^{(t)}$.*

Proof. First, we prove $UB^{(t)} = U^{(t)}$. Due to the strictly convex property, P1 has the strong duality, by which we have P1 which is equivalent to

$$\begin{aligned}
\max_{(\mathbf{X})} \quad & \left\{\min_{\boldsymbol{\nu},\boldsymbol{\omega},\boldsymbol{\gamma},\boldsymbol{\mu}} \left\{\sup_{(\boldsymbol{\Lambda})} \left\{\Gamma\left(\mathbf{X}_{\mathcal{J}}, \mathbf{X}_{\mathcal{K}}, \boldsymbol{\Lambda}_{\mathcal{J}}, \boldsymbol{\nu}\right)\right\}\right\}\right\}, \\
\text{s.t.} \quad & \sum_{l\in\mathcal{F}\cup\mathcal{J}} x_{il} = 1, \\
& \sum_{i\in\mathcal{J}} x_{ik} \le N_m, \\
& x_{il} \in \{0,1\}, \\
& i \in \mathcal{F}, \\
& l \in \mathcal{F}\cup\mathcal{J}.
\end{aligned} \tag{27}$$

Therefore, problem P1 and (27) have the same optimal solution and objective value of goal function. Denote the optimal solution of P1 as $(\mathbf{X}^*, \boldsymbol{\Lambda}^*)$ and the corresponding goal function value as $\Phi(\boldsymbol{\Lambda}^*, \mathbf{X}^*) = (\sum_i \sum_l (x_{il}^* \log(\sum_l R_{il}\lambda_{il}^* + \beta_l))) = M^*$. In tth iteration, the optimal solution to problem (26) is $(\mathbf{X}^{(t)}, \eta^{(t)})$. Since (26) is a relaxation of (27), it follows that $\Phi(\boldsymbol{\Lambda}^*, \mathbf{X}^*) \le U^{(t)} = \eta^{(t)}$. Therefore, $U^{(t)}$ is a upper bound of the objective function in P1.

Next, we prove $LB^{(t)} = \max_{0\le s\le t}\{L^{(s)}\}$ and $L^{(s)} = \Phi(\boldsymbol{\Lambda}^{(s)}, \mathbf{X}^{(s)})$ is an upper bound of P1. Whether $L^{(t)}$ is bounded is decided by (24). If $L^{(s)} = -\infty$ $(\forall 0 \le s \le t)$, $LB^{(t)} = \max_{1\le s\le t}\{\Phi(\boldsymbol{\Lambda}^{(s)}, \mathbf{X}^{(s)})\} = -\infty$, which is obviously the lower bound of P1, so we focus on the case in which problem (24) is bounded. If (24) is bounded, $LB^{(t)} > -\infty$. We denoted $\underline{s} = \arg\max_{1\le s\le t}\{\Phi(\boldsymbol{\Lambda}^{(s)}, \mathbf{X}^{(s)})\}$, and thereby $LB^{(t)} = L^{(\underline{s})} = \Phi(\boldsymbol{\Lambda}^{(\underline{s})}, \mathbf{X}^{(\underline{s})})$. If $LB^{(t)}$ is not the lower bound of P1, $LB^{(t)} > M^*$. Due to the strong duality of problem (24), $L^{(\underline{s})} = \Phi(\boldsymbol{\Lambda}^{(\underline{s})}, \mathbf{X}^{(\underline{s})}) > M^* = \Phi(\boldsymbol{\Lambda}^*, \mathbf{X}^*)$. This means $(\boldsymbol{\Lambda}^{(\underline{s})}, \mathbf{X}^{(\underline{s})})$ generates a larger objective function value of P1 than that $(\boldsymbol{\Lambda}^*, \mathbf{X}^*)$ does, which is in contradiction to the fact that $(\boldsymbol{\Lambda}^*, \mathbf{X}^*)$ is the optimal solution of P1. Therefore, $LB^{(t)} = \max_{0\ge s\ge t}\{L^{(s)}\}$ is a lower bound of P1. □

Theorem 2. *The ASRAO algorithm converges to a global optimal solution to P1 with finite number of iterations.*

Proof. As is indicated in [19, 20], there always exists a pointed convex polyhedral cone $\mathscr{C}$ that makes problem (24) and (26) equivalent to

$$\begin{aligned}
\max_{(\mathbf{X},\eta)} \quad & \eta, \\
\text{s.t.} \quad & (\mathbf{X}, \eta) \in \mathscr{C}, \\
& \sum_{l\in\mathcal{F}\cup\mathcal{K}} x_{il} = 1, \\
& \sum_{i\in\mathcal{J}} x_{ik} \le N_m, \\
& x_{il} \in \{0,1\}, \\
& \forall i \in \mathcal{F}, \\
& \forall k \in \mathcal{K}, \\
& \forall l \in \mathcal{J}\cup\mathcal{K},
\end{aligned} \tag{28}$$

wherein $\mathscr{C}$ can be expressed as a convex hull of finitely many half-lines. In each iteration of ASRAO the set of solution $(\mathbf{X}, \eta)$ to problem (26) is shrunk by introducing a new extreme half-line being either an optimality cut $\Gamma(\mathbf{X}, \boldsymbol{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}) \ge \eta$ or a feasibility cut $\sum_{(i,l)\in Y_v} x_{il} - \sum_{(i,l)\in N_v} x_{il} \le |Y_v| - 1$. Because $\mathscr{C}$ is a convex hull of finally many extreme half-lines and the new introduced extreme half-line is different from the preceding ones, the complete set of constraints determining the $\mathscr{C}$ can be obtained within a finite number of iterations. This means the optimal UE-AN associations $\mathbf{X}^*$ and solution to problem (24) can be obtained within finite iterations by solving

$$\Phi\left(\boldsymbol{\Lambda}^{(u)}, \mathbf{X}^{(t)}\right) = \Gamma\left(\mathbf{X}^{(t)}, \boldsymbol{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}\right). \tag{29}$$

Therefore, the optimal UE-AN associations $\mathbf{X}^*$ and solution to problem P1 can also be obtained within a finite number of

iterations. As is analyzed above, the sequence $\{UB^{(t)}\}$ is nonincreasing, and after the set $\mathscr{C}$ is determined by constraints in (29), $UB^{(k)} = M^*$, where M^* denotes the optimal objective function value. The sequence $\{LB^{(t)}\}$ is nondecreasing and satisfies $LB^{(t)} = M^*$ after $\mathbf{X}^*$ is found through solving (29). Therefore, we can claim that $UB^{(t)} - LB^{(t)} = 0$ will guarantee that the ASRAO converges to the optimal solution to problem P1 within a finite number of iterations. □

3.4. Accelerated ASRAO Algorithm. Problem (26) is an MILP, which is generally calculated by branch and bound approach. Thus, the computation of problem (24) dominates the computation complexity of the whole BD process. In order to reduce the cost of solving problem (26), in this paper we propose to accelerate ASRAO algorithm by relaxing integer constraints in (26) into continuous constraints in intermediate iterations. Specifically, in tth iteration, the MP can be relaxed into a LP problem, as is in

$$\begin{aligned}
\max_{(\mathbf{X}_{\mathscr{J}},\mathbf{X}_{\mathscr{K}},\eta)} \quad & \eta, \\
\text{s.t.} \quad & \Gamma\left(\mathbf{X}, \boldsymbol{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}\right) \geq \eta, \\
& \qquad \forall u = 1, \ldots, t_1, \\
& \sum_{(i,l)\in Y_v} x_{il} - \sum_{(i,l)\in N_v} x_{il} \leq \left|Y_v\right| - 1, \\
& \qquad \forall v = 1, \ldots, t_2, \\
& \sum_{l\in\mathscr{F}\cup\mathscr{J}} x_{il} = 1, \\
& \sum_{i\in\mathscr{J}} x_{ik} \leq N_m, \\
& x_{il} \in [0,1], \\
& i \in \mathscr{J}, \\
& k \in \mathscr{K}, \\
& l \in \mathscr{F}\cup\mathscr{J},
\end{aligned} \tag{30}$$

where x_{il} is a relaxed continuous variable ranged in $[0, 1]$. As problem (30) is a LP problem, it can be solved utilizing standard algorithms, simplex method for instance. Denote the solution derived from problem (30) as $(\widehat{\mathbf{X}}, \hat{\eta})$, in which the entries in $\widehat{\mathbf{X}}$ are possibly not integers. Theorem 3 addresses the fact that relaxation of problem (26) will not exclude the optimal solution to P1.

Theorem 3. *Denote $(\widehat{\mathbf{X}}, \hat{\eta})$ as an arbitrary feasible solution to the relaxed problem. The optimality cuts and feasibility cuts generated with $\widehat{\mathbf{X}}$ will not exclude the optimal solution $(\boldsymbol{\Lambda}^*, \mathbf{X}^*)$ from problem P1.*

Proof. If problem (24) is bounded, denote $(\widehat{\boldsymbol{\Lambda}}, \widehat{\boldsymbol{\nu}})$ as the optimal solution to problem (24) with $\widehat{\mathbf{X}}$ and an optimality cut $\Gamma(\mathbf{X}, \widehat{\boldsymbol{\Lambda}}_{\mathscr{J}}, \widehat{\boldsymbol{\nu}}, \widehat{\boldsymbol{\omega}}, \widehat{\boldsymbol{\gamma}}, \widehat{\boldsymbol{\mu}}) \geq \eta$ is generated. Otherwise, a feasibility cut $\sum_{(i,l)\in Y_v} x_{il} - \sum_{(i,l)\in N_v} x_{il} \leq |Y_v| - 1$ is generated. To prove Theorem 3, we show that, in either cases, optimal solution $(\boldsymbol{\Lambda}^*, \mathbf{X}^*)$ to problem P1 will not violate the newly introduce constraints. Hence, it will not be excluded from the feasible set by the generated cuts.

Denote the optimal solution to problem (24) as $(\boldsymbol{\Lambda}^*, \boldsymbol{\nu}^*, \boldsymbol{\omega}^*, \boldsymbol{\gamma}^*, \boldsymbol{\mu}^*)$ with $\mathbf{X}^*$; the corresponding optimal value of the objective function is $M^* = \eta^* = \Phi(\boldsymbol{\Lambda}^*, \mathbf{X}^*)$. In the case where problem (24) is bounded with $\mathbf{X}$ being $\widehat{\mathbf{X}}$, suppose $(\mathbf{X}^*, \eta^*)$ violates $\Gamma(\mathbf{X}, \widehat{\boldsymbol{\Lambda}}, \widehat{\boldsymbol{\nu}}, \widehat{\boldsymbol{\omega}}, \widehat{\boldsymbol{\gamma}}, \widehat{\boldsymbol{\mu}}) \geq \eta$, that is $\Gamma(\mathbf{X}^*, \widehat{\boldsymbol{\Lambda}}, \widehat{\boldsymbol{\nu}}, \widehat{\boldsymbol{\omega}}, \widehat{\boldsymbol{\gamma}}, \widehat{\boldsymbol{\mu}}) < \eta^*$, due to the completeness of the constraints in (26), it means a smaller $\eta' = \Gamma(\mathbf{X}^*, \widehat{\boldsymbol{\Lambda}}, \widehat{\boldsymbol{\nu}}, \widehat{\boldsymbol{\omega}}, \widehat{\boldsymbol{\gamma}}, \widehat{\boldsymbol{\mu}})$ is found, which violates the fact that $M^* = \eta^*$ is the optimal solution to problem P1. Hence, the optimality cut will not be violated. In the case where the problem (24) is unbounded, suppose $\mathbf{X}^*$ violates the feasibility cut $\sum_{(i,l)\in Y_v} x_{il} - \sum_{(i,l)\in N_v} x_{il} \leq |Y_v| - 1$, which means $\mathbf{X}^*$ makes problem (24) unbound, and therefore, $\mathbf{X}^*$ will not be the optimal solution to P2. This conflicts the fact that $(\mathbf{X}^*, \eta^*)$ is the optimal solution to P1. So the feasibility cut will not be violated.

Theorem 3 means we can safely replace (26) by (30) in our ASRAO algorithm without concerning about losing the optimal solution of P1. However, it should be noted that $\widehat{\mathbf{X}}$ in the optimal solution to problem (30) can be used to calculate the lower bound of P1, but when problem (24) is solved with $\widehat{\mathbf{X}}$, the upper bound derived from the objective function value of problem (24) may not be a valid upper bound for P1, since entries in $\widehat{\mathbf{X}}$ are not integers and thereby infeasible to constraints in P1. One way to cope with the problem is to round $\widehat{\mathbf{X}}$ to the nearest matrix $\widetilde{\mathbf{X}}$ with binary entries. However, as the summation of elements in each row of $\mathbf{X}$ equals 1, a direct approximation would produce an all-zero matrix. Hence, in view of the special property of $\mathbf{X}$ that the entries in the matrix are binary variables and only one element in each row equals 1, we use a technique inspired by feasible pump method proposed in [21] to get $\widetilde{\mathbf{X}}$.

To select a 1 element in x_i, the ith row of $\widehat{\mathbf{X}}$, we divide a range $[0, 1]$ into $J + K$ slots, and each element in x_i successively occupies a slot, whose length equals to the value of element x_{il}. Then a randomizer uniformly generates a pointer valued within $[0, 1]$. This element with the slot where the pointer locates is selected, and all other elements are 0 elements. To control randomness, a parameter T is used to decide only $1/T$ rows are allowed to perform randomly selection and other rows just select their largest elements as 1 element.

Thereby, from $\widehat{\mathbf{X}}$ we get an approximate matrix $\widetilde{\mathbf{X}}$ which is possibly feasible to problem (26) and P1, but with very low probability $\widetilde{\mathbf{X}}_{\mathscr{K}}$ fails the constraint $\sum_{i\in\mathscr{J}} x_{ik} \leq N_m$, and therefore, $\widetilde{\mathbf{X}}_{\mathscr{K}}$ should be checked. And to avoid the fact that $\widetilde{\mathbf{X}}$ stays at a fixed point, novelty of the solution should be checked. If no feasible solution or solution stays unchanged, we revert back to the normal ASRAO algorithm, and the current solution calculated from problem (30) serves as an initial $\widetilde{\mathbf{X}}$ to the following normal ASRAO process.

The pseudocodes of accelerated ASRAO algorithm are as shown in Algorithm 2. □

(1) **Initialize**: $UB^{(0)} = +\infty$, $LB^{(0)} = -\infty$, Let $t = 0$, $u = 0$, $v = 0$.
PHASE-I
(2) **repeat**
(3) Let $t = t + 1$ and solve (30) to obtain current optimal solution $(\widehat{\mathbf{X}}^{(t)}, \widehat{\eta}^{(t)})$, $UB^{(t)} = \widehat{\eta}^{(t)}$
(4) **if** $\widehat{\mathbf{X}}^{(t)} = \widehat{\mathbf{X}}^{(t-1)}$ or feasible $\widetilde{\mathbf{X}}^{(t)}$ can not be found: $t = t - 1$, **break**
(5) **else**: get approximative matrix $\widetilde{\mathbf{X}}^{(t)}$, and $\mathbf{X}^{(t)} = \widetilde{\mathbf{X}}^{(t)}$
(6) **if** (24) with $\mathbf{X}^{(t)}$ is bounded: $u = u + 1$, and solve it with $\mathbf{X}^{(t)}$ to obtain KKT point $(\boldsymbol{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)})$ $= (\boldsymbol{\Lambda}^{(t)}, \boldsymbol{\nu}^{(t)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(t)})$. The lower bound is set to $LB^{(t)} = \max\{\max_{1 \le s \le t}\{\Phi(\boldsymbol{\Lambda}^{(s)}, \mathbf{X}^{(s)})\}, LB^{(0)}\}$.
Add $\Gamma(\mathbf{X}, \boldsymbol{\Lambda}^{(u)}, \boldsymbol{\nu}^{(u)}, \boldsymbol{\omega}^{(u)}, \boldsymbol{\gamma}^{(u)}, \boldsymbol{\mu}^{(u)}) \ge \eta$ to (26) and (30)
(7) **else**: $v = v + 1$, $\mathbf{X}^{(v)} = \mathbf{X}^{(t)}$, $LB^{(t)} = LB^{(t-1)}$, and add $\sum_{(i,l)\in Y_v} x_{il} - \sum_{(i,l)\in N_v} x_{il} \le |Y_v| - 1$ to (26) and (30).
(8) **until** $UB^{(t)} - LB^{(t)} \le 0$
PHASE-II
(9) ASRAO step (2) to (7).
(10) Return $(\mathbf{X}^*, \boldsymbol{\Lambda}^*) = (\mathbf{X}^{(t)}, \boldsymbol{\Lambda}^{(t)})$ as the optimal solution to problem P1.

Algorithm 2: The A-ASRAO algorithm.

4. Performance Evaluation

In this section, we first introduce network scenarios and parameter settings used in our simulations. After this, the performance of ASRAO is evaluated in scenarios with different AP and UE settings. The effectiveness of the accelerative technique is also discussed here. Then, we analyze effects the value of β may cause to our UTMAX model by evaluating the optimized network performances on metrics, such as system throughput and access fairness. In the last part of this section, our UTMAX model is evaluated under different AP densities to validate its effectiveness and compatibility.

4.1. Scenario and Parameter Settings. The simulations are carried out in a scenario consists of 2 fixed BSs and 6 or 12 APs. The distance between the two BSs is about 300 m. The number of UEs varies from 10 to 20 by a step size of 2. In order to reduce random errors, each point of the simulation results is averaged over 100 different network topologies among which the positions of APs and UEs differ but their amounts stay unchanged. According to [15], we simplify the propagation loss as $37 + 37.6 \log d$ (km) + 21 log (2.6/2) dB in CN, and $34 + 37.6 \log d$ (km) + 21 log (2.4/2) dB in WLAN. To model the shadowing effects in CN a log-normal random variable with zero mean and 8 db variation is introduce in CN, while in WLAN shadowing effects are neglected. Moreover, a received power maximization (PRMAX) user association scheme, which is commonly used in present commercial networks, serves as a performance benchmark. Another scheme maximally uses WLAN by getting AP to serve each possible user in its range. We denote it as APMAX. Our UTMAX optimization model with different β values for different system preferences, such as system throughput and user data rate fairness, is compared with PRMAX and APMAX to validate its performance. The main parameters and their values are as listed in Table 1, from [7, 15].

4.2. Algorithm Performance Analyses. In this subsection, we validate the convergence of ASRAO and A-ASRAO in simulations and analyze the effects the value of a randomness control parameter T produces on the convergence of the latter. The simulations are carried out in Matlab under the settings mentioned above, and our algorithms are implemented with YALMIP [22], an optimization toolbox, which is able to solve our UTMAX model directly, but at unsatisfying speed. We denote it by B&B, because an MINLP solver based on branch and bound is used. It works differently in each detailed iteration with (A-)ASRAO, so only the computation time spent by these approaches is compared.

Table 1: Simulation parameter.

Items (unit)	Values
Transmitting power of BSs (dBm)	46
Transmitting power of WiFi AP (dBm)	20
Distance between BSs (m)	300
LTE bandwidth (MHz)	10
WiFi bandwidth (MHz)	20
eNB antenna height from eNB (m)	40
AP antenna height from AP (m)	5
LTE operating frequency (MHz)	2600
WiFi operating frequency (MHz)	2400
Number of eNode BSs and APs	2 : 6, 2 : 12
Thermal noise (dBm/Hz)	−174
Shadowing for LTE (dB)	8
LTE cochannel interference coefficient α	1
Number of UEs	10 : 2 : 20
Min resource for UE in eNB λ_ε	0.01
Max resource for UE in AP λ_m	0.3
Fairness coefficient β	−0.05, 0, 1, 5

Figure 2 illustrates the convergent processes of ASRAO and A-ASRAO in a UTMAX instance where 20 UEs exist. In essence, ASRAO performs fixed path searching, whose interim solutions are decided by their preceding solutions. ASRAO is sure to converge, but it lacks the ability to try better searching paths. While A-ASRAO adopts a random

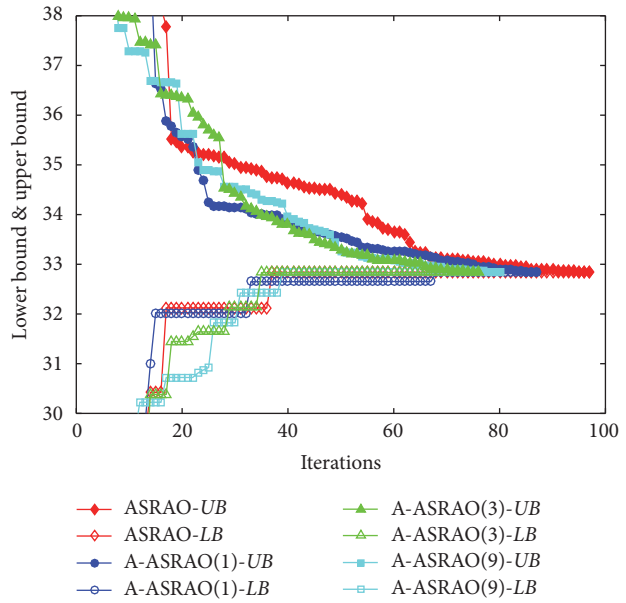

FIGURE 2: The converge process of ASRAO and A-ASRAO with $T = (1, 3, 9)$.

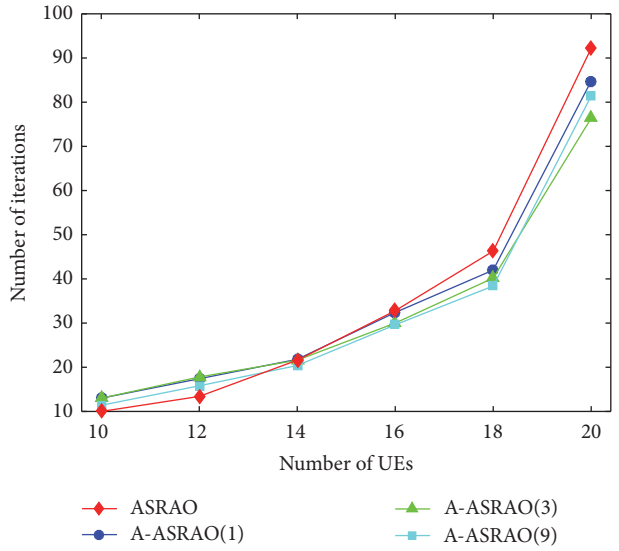

FIGURE 3: The number of iterations versus the number of UEs.

searching strategy, with wider searching directions, it shows more steady descending upper bound. From Figure 2, it is obvious that A-ASRAO converges much faster in the later part of the iterations than ASRAO as is depicted by lines with hollow marks. As these upper bounds reach the same optimal objective function value and all lower bounds get to optima much earlier than upper bounds, A-ASRAO will converge much faster than ASRAO.

In addition, A randomness control parameter T is set in A-ASRAO to control its behavior, and A-ASRAO with T is denoted by A-ASRAO(T). With different values of T, A-ASRAO has different converging rates. This is probably caused by random generation of approximation matrix $\widetilde{\mathbf{X}}$. Smaller T means more rows in $\widetilde{\mathbf{X}}$ perform random 1-element selection, while large T will make less rows perform random selection. Extremely small or large values of T will make too many or few rows perform random selection, but neither of them is good for finding ideal initial solution for Phase II in A-ASRAO. So A-ASRAO(3) performs better than A-ASRAO(1) and A-ASRAO(9).

In Figures 3 and 4, respectively, the number of iterations performed and computation time spent by ASRAO and A-ASRAO are plotted against the number of UEs ranging from 10 to 20. Each point is obtained by averaging over 100 different network topologies. It shows that ASRAO needs a little fewer iterations than A-ASRAO when UE number is less than 14. However, when UE number is larger, A-ASRAO reduces the number of iterations by up to 15% compared with ASRAO in average.

The reason for the different convergent properties lies in the scale of a UTMAX instance. A-ASRAO works in two phases and in Phase II A-ASRAO works the same as ASRAO, which implies that A-ASRAO in Phase I works actually to obtain an initial solution to Phase II in A-ASRAO. That is to say, if a better solution, which produces smaller gap between the upper bound and lower bound, found by the end of Phase I in A-ASRAO than that by ASRAO within the same iterations, A-ASRAO will have much larger probability to converge within fewer iterations than ASRAO. Therefore, the dominant factor relating to the different convergent rates of ASRAO and A-ASRAO is whether a better solution would be found by the end of Phase I in A-ASRAO.

When a UTMAX instance is small-scale problem, its optimal solution can be derived within a few iterations even through exhaustive search, not alone ASRAO. While A-ASRAO tries a more steady but random searching, the finding of optimal solution may thus be postponed. So it hardly performs better than ASRAO. When a UTMAX instance is of large scale, a large solution space means a bolder searching practice may produce better searching path. So A-ASRAO(3) performs better than ASRAO and A-ASRAO(9). Nevertheless, too strong randomness goes against the finding of optimal solution, like in A-ASRAO(1) where all rows in $\widetilde{\mathbf{X}}$ perform random 1-element selection. We speculate that is because a too wild searching will waste many iterations wandering around rather than heading towards a converging direction. That is why A-ASRAO(3) outperforms A-ASRAO(1). Moreover, in Figure 4 it shows that A-ASRAO spends less time computing optimal solution than ASRAO, and the performance gap is even larger than iteration numbers. This benefits from relaxing the MILP master problem into LP in Phase I of A-ASRAO, and LP can be solved much faster than MILP. Figure 4 also shows that ASRAO computes UTMAX much quicker than a common B&B method. Above results prove the effectiveness of (A-)ASRAO in solving UTMAX, and in most cases A-ASRAO(3) performs better than others.

4.3. Effect of β to UTMAX. As is mentioned in Section 2, β indicates UTMAX's preference on throughput fairness among UEs served in a BS by setting effects on resource

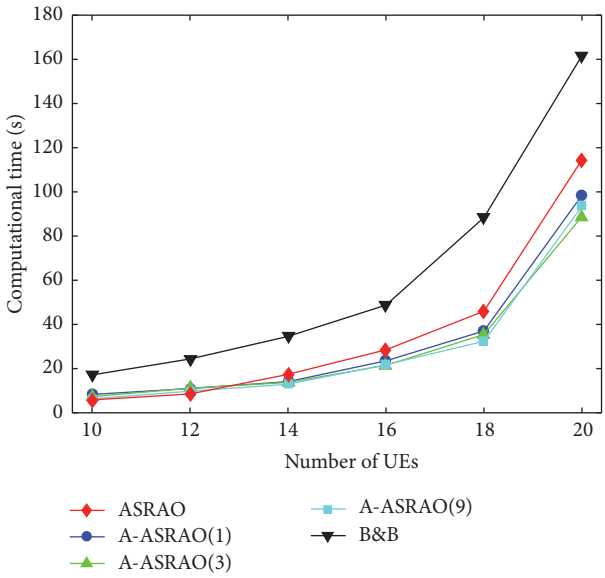

Figure 4: Computational time versus the number of UEs.

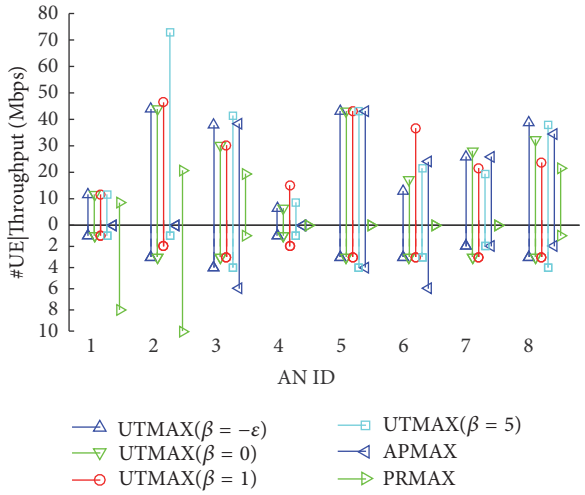

Figure 5: Performance at AN granularity of PRMAX, APMAX, and UTMAX(β).

allocation. As UE's attainable data rate is changed, user associations are inevitably affected, which is reflected in Figure 5. And as a consequence, the performance of the entire interworking network is influenced. To analyze these effects, the performance of optimized network is evaluated on throughput produced and the number of UEs served in WiFi tier and cellular tier in Figure 6 and throughput gains of UEs in Figure 7 through APMAX and UTMAX with β in $\{-0.05, 0, 1, 5\}$. PRMAX is a default user association scheme and adopts identical resource allocation in each BS and AP.

Figure 5 plots the throughput and number of UEs served in each BS and AP with different approaches marked by different colors and symbols. It shows that, from a AN perspective, both APMAX and UTMAX do well in improving AP's utilization, and APMAX performs even better, as it tends to maximize AP's usage. The improvements of WLAN's

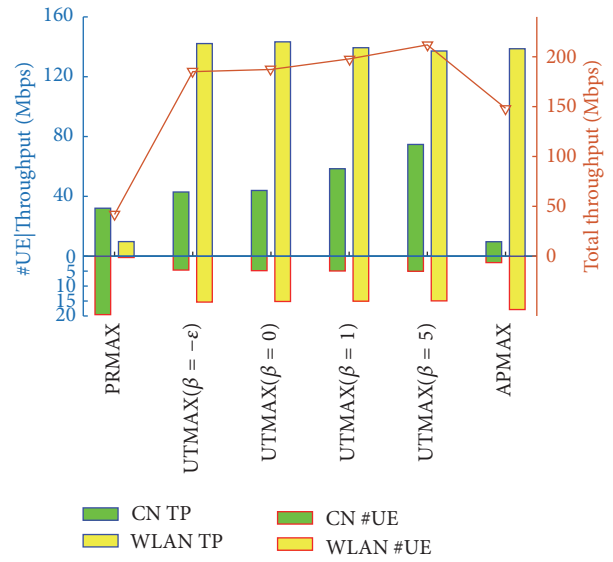

Figure 6: Performance at network granularity of PRMAX, APMAX, and UTMAX(β).

offloading performance by different approaches are much more obvious in Figure 6.

Figure 6 plot the throughput and number of UEs served in cellular tier and WiFi tier. APMAX gets most UEs served in WLAN, while, as resources are fully occupied, each UE in WLAN will get less resources and the throughput of WLAN will not be improved greatly compared with UTMAX. However, due to lack of careful UE selection from CN to WLAN in APMAX, UE amount decreases greatly in CN and the left UEs are probably with poor wireless channel condition or too far from a BS, which causes the poor CN throughput performance in APMAX. As for UTMAX, with different β value, it performs differently. Larger β value will get BS to allocate more resources to UEs with higher unit band data rate, which probably makes the contribution to the throughput growing in CN. However, as UEs with poor channel condition in CN get much less resources, they turn to WLAN to compensate the throughput loss. That would cause a minor UE amount increase in WLAN, while, as AP's resources are equally shared, new coming UEs with poor channel condition share resources with indigenous WLAN UEs, causing the decrease in WLAN's throughput. Nevertheless, the whole system throughput shows a continuous growth with the increase of β. It seems that larger β performs better on total throughput and throughput fairness between WLAN and CN, whereas it is on the expense of throughput fairness among UEs.

Figure 7 depicts UE throughput gain versus probability $\mathbb{P}(r < \alpha)$ for various approaches versus PRMAX, where $\mathbb{P}(r < \alpha)$ represents the ratio of UE whose throughput is less than α in all UE. The UEs' throughput gains are quite large at 10% ratio point by all approaches, among which UTMAX($-\varepsilon$) brings the largest throughput gain. With the growing of β, UE throughput gain decreases. However, UTMAX always has higher throughput gain over APMAX. With the growth of probability $\mathbb{P}(r < \alpha)$ UE throughput gain decreases,

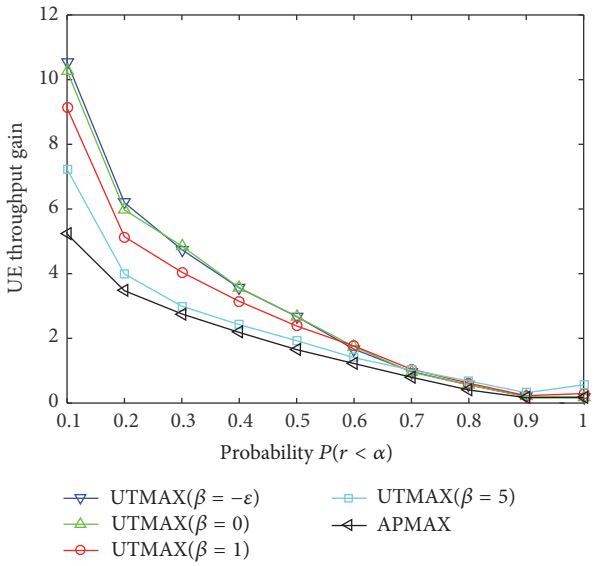

Figure 7: Performance at UE granularity of PRMAX, APMAX, and UTMAX(β).

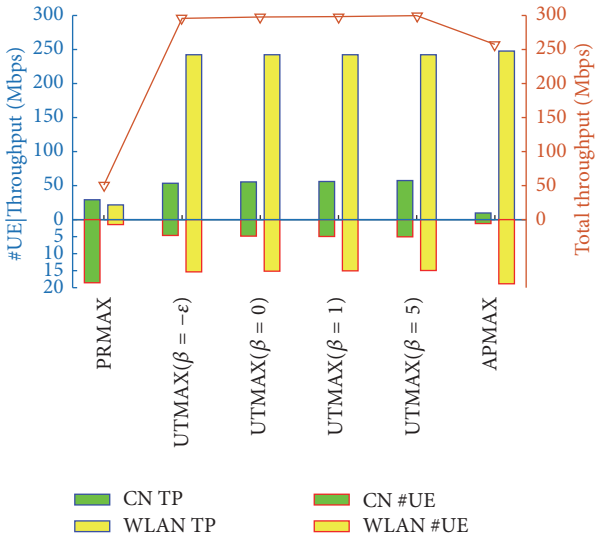

Figure 8: Performance at network granularity of PRMAX, APMAX, and UTMAX(β).

because PRMAX also has relatively larger UE throughput at larger ratio points. At the meanwhile, UTMAX with larger β value has larger UE throughput gain, as larger β makes UTMAX allocate more resources to UEs with better channel conditions.

4.4. Effects of AP Density to UTMAX. The performances of UTMAX with different values of β are evaluated on larger AP density, to validate whether the same conclusion holds with different AP settings. The results are in Figures 8 and 9.

In Figure 8, compared with Figure 6 the most obvious changes are the WLAN tier throughput increase for PRMAX and minimization of performance gap between UTMAXs with each of two different values of β. Due to the increase of

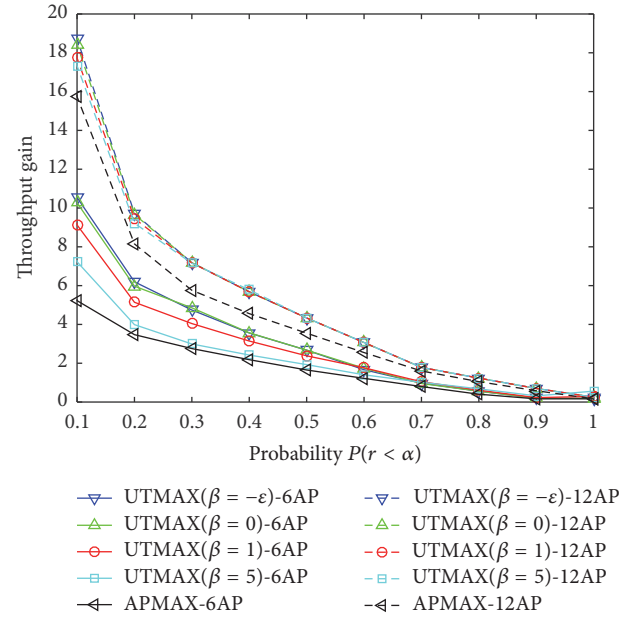

Figure 9: Performance at UE granularity of PRMAX, APMAX, and UTMAX(β).

AP density, more UEs access WLAN and the average distance from a UE to an AP is shorten. This is a direct cause to WLAN tier throughput increase for PRMAX and APMAX. As for the latter difference, it is possibly because of much fewer UEs in each BS or AP, and β probably has weaker effect on resource allocation among fewer UEs in a serving node. Figure 9 is plotted by adding curves of UE throughput gain where 12 APs exist. The additional UE throughput gain is largely contributed by newly added APs. A similar performance relation is shown between UTMAX with larger AP density, while the performance gap is narrowed and the curves meet and overlap earlier than that with smaller AP density.

From these results, the effect of β to UTMAX is further checked that a smaller β makes UTMAX perform better on improving throughput of the part of UEs with poor channel conditions, while a larger β does better on improving overall system throughput.

5. Conclusion

To solve a key research case in multi-RAT SON about alleviating the biased load distribution in cellular/WLAN interworking network, we propose to optimize access load from users satisfaction perspective, and a UTMAX optimization model with sum of logarithmic utilities is derived. This is an MINLP and seems hard to solve intuitively. In order to solve it optimally and efficiently, we devise an ASRAO algorithm on the basis of general Benders Decomposition and prove its convergence in finite iterations to optimal. To reduce the computation complexity in ASRAO, especially when dealing with large-scale problem instance, we propose to accelerate the ASRAO algorithm with a relaxation and approximation technique inspired by feasible pump. Simulation results validate the convergence and effectiveness of both ASRAO

and A-ASRAO and prove the effects of UTMAX optimization model on improving throughput fairness among users or the system throughput, and in either case, the utilization of WLAN is improved obviously especially on the number of users and throughput in it.

As shown in simulation results, when dealing with large-scale UTMAX instance, A-ASRAO has distinct advantage over ASRAO on convergent speed, which implies that A-ASRAO is a proper algorithm to solve UTMAX. The resource allocation among users in BS is affected by β, whereas simulation results imply it, and also functions as an indicator for preference on throughput fairness among users or total system throughput, which is a tradeoff or preference from viewpoints of network operators. However, due to computation complexity limitation, it is hard to perform extensive simulations on much larger UTMAX instances. So our future work includes a heuristic approach to produce solution close to that of ASRAO but with much lower computational complexity, and therefore effects of UTMAX can be further checked with larger UE density.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work is supported by National High-Tech Research and Development Program of China (2015AA01A705) and National Natural Science Foundation of China (61271187).

References

[1] R. Litjens, "D6.6 final report on a unified self-management system for heterogeneous radio access networks," Tech. Rep. INFSO-ICT-316384, SEMAFOUR, 2015.

[2] D. Laselva, "D4.1 SON functions for multi-layer LTE and multi-RAT networks (first results)," Tech. Rep. INFSO-ICT-316384, SEMAFOUR, 2013.

[3] F. Rebecchi, M. Dias de Amorim, V. Conan, A. Passarella, R. Bruno, and M. Conti, "Data offloading techniques in cellular networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 580–603, 2015.

[4] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting mobile 3G using WiFi," in *Proceedings of the 8th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '10)*, pp. 209–222, ACM, San Francisco, Calif, USA, June 2010.

[5] M. D. Nisar, V. Pauli, and E. Seidel, "Multi-RAT traffic steering—why, when, and how could it be beneficial?" Whitepaper, 2011.

[6] D. H. Hagos, *The performance of WiFi offload in LTE networks [M.S. thesis]*, Lulea University of Technology, Lulea, Sweden, 2012.

[7] I. Balan, D. Laselva, S. Redana, and A. Lobinger, "RSRP-based LTE-WLAN traffic steering," in *Proceedings of the 81st IEEE Vehicular Technology Conference (VTC '15)*, pp. 1–5, May 2015.

[8] S. Shin, D. Han, H. Cho, and J.-M. Chung, "Improved association and disassociation scheme for enhanced WLAN handover and VHO," *Mobile Information Systems*, vol. 2016, Article ID 4868479, 6 pages, 2016.

[9] S. J. Bae, M. Y. Chung, and J. So, "Handover triggering mechanism based on IEEE 802.21 in heterogeneous networks with LTE and WLAN," in *Proceedings of the International Conference on Information Networking (ICOIN '11)*, pp. 399–403, January 2011.

[10] K. Piamrat, A. Ksentini, C. Viho, and J.-M. Bonnin, "QoE-aware vertical handover in wireless heterogeneous networks," in *Proceedings of the 7th International Wireless Communications and Mobile Computing Conference (IWCMC '11)*, pp. 95–100, Istanbul, Turkey, July 2011.

[11] A. I. Aziz, S. Rizvi, and N. M. Saad, "Fuzzy logic based vertical handover algorithm between LTE and WLAN," in *Proceedings of the International Conference on Intelligent and Advanced Systems (ICIAS '10)*, pp. 1–4, IEEE, Kuala Lumpur, Malaysia, June 2010.

[12] K. Premkumar and A. Kumar, "Optimum association of mobile wireless devices with a WLAN-3G access network," in *Proceedings of the IEEE International Conference on Communications (ICC '06)*, vol. 5, pp. 2002–2008, Istanbul, Turkey, July 2006.

[13] J. Chen, L. P. Qian, and Y. J. Zhang, "On optimization of joint base station association and power control via Benders' decomposition," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '09)*, pp. 1–6, IEEE, Honolulu, Hawaii, USA, December 2009.

[14] S. Kim, S. Choi, and B. G. Lee, "A joint algorithm for base station operation and user association in heterogeneous networks," *IEEE Communications Letters*, vol. 17, no. 8, pp. 1552–1555, 2013.

[15] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. G. Andrews, "User association for load balancing in heterogeneous cellular networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2706–2716, 2013.

[16] S. Singh, S. Yeh, N. Himayat, and S. Talwar, "Optimal traffic aggregation in multi-RAT heterogeneous wireless networks," in *Proceedings of the IEEE International Conference on Communications Workshops (ICC '16)*, pp. 626–631, Kuala Lumpur, Malaysia, May 2016.

[17] T. GPP, "23.234 v6. 2.0, 3gpp system to wireless local area network (wlan) interworking," System description (Release 6), 2004.

[18] "IEEE Standard for Local and metropolitan area networks—Media Independent Handover Services," IEEE Std 802.21-2008, 2009.

[19] J. F. Benders, "Partitioning procedures for solving mixed-variables programming problems," *Numerische Mathematik*, vol. 4, pp. 238–252, 1962.

[20] A. M. Geoffrion, "Generalized Benders decomposition," *Journal of Optimization Theory and Applications*, vol. 10, no. 4, pp. 237–260, 1972.

[21] M. Fischetti, F. Glover, and A. Lodi, "The feasibility pump," *Mathematical Programming*, vol. 104, no. 1, pp. 91–104, 2005.

[22] J. Löfberg, "YALMIP: a toolbox for modeling and optimization in MATLAB," in *Proceedings of the IEEE International Symposium on Computer Aided Control System Design*, pp. 284–289, September 2004.