This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2018.2810228, IEEE Transactions on Mobile Computing

Optimal Resource Allocations for Mobile Data Offloading via Dual-Connectivity

Yuan Wu *Senior Member IEEE*, Yanfei He, Li Ping Qian *Senior Member IEEE*,
Jianwei Huang *Fellow IEEE*, Xuemin (Sherman) Shen *Fellow IEEE*

Abstract—**The rapid growth of mobile traffic has heavily overloaded the cellular networks, making it increasingly desirable to offload mobile users' (MUs') traffic to small-cell networks. In this paper, we study the MUs' optimal uplink traffic offloading scheme based on the new paradigm of small-cell dual-connectivity (DC). Through DC, an MU can flexibly schedule its traffic between a macro-cell base station (BS) and a small-cell access point (AP) via two different radio interfaces. To optimize the overall network radio resource usage, we jointly optimize the BS' bandwidth allocation as well as the MUs' traffic scheduling and power allocation. Specifically, for reducing the bandwidth usage, the BS prefers to allocate the MUs small amount of bandwidth to encourage the MUs to utilize the small-cell networks. However, excessive traffic offloading can lead to severe interferences among MUs, which increase the MUs' power consumption. Hence our joint optimization strikes a proper balance between these two aspects. Despite the non-convexity of the proposed joint optimization problem, we propose an efficient algorithm to compute the optimal offloading solution. The key idea is to exploit the layered-structure of the joint optimization problem, and decompose it into the BS' bandwidth allocation problem (on the top-level) and the MUs' traffic scheduling and power allocation problem (as a subproblem). Such a decomposition enables us to exploit the hidden convexity of the MUs' problem and the monotonic structure of the BS' problem for an effective algorithm design. Numerical results show that our proposed algorithm can achieve the global optimum solution with significantly reduced computational time. Moreover, the proposed traffic offloading scheme can significantly reduce the overall system cost, in comparison with using the fixed bandwidth allocation or traffic scheduling schemes.**

Index Terms—**Traffic Offloading, Dual Connectivity, Small-cell Networks, Resource Optimization, Non-convex Optimization**

I. Introduction

The proliferation of smart and media-hungry mobile services have led to a rapid growth of mobile data traffic, which have increasingly overloaded cellular networks and degraded mobile users' (MUs') quality of services (QoS) [1]. How to effectively serve such heavy growing demands in a timely and cost-efficient manner becomes critically challenging to network operators. Thanks to the widely deployed small-cell networks (such as Wi-Fi access points and femtocells) that provide access capacities, offloading MUs' traffic to small-cell networks becomes a promising approach to reduce the stress on the cellular macro-cell networks [2] [3]. Due to the typical close-proximity between MUs and small-cell networks, traffic offloading can bring several benefits, such as saving radio resources, improving users' quality of services, and reducing users' mobile data costs. However, aggressive data offloading from multiple MUs to the same small-cell may cause severe interference. Therefore, it is important to design a proper radio resource management scheme for mobile data offloading.

Very recently, a new paradigm of *small-cell dual-connectivity* (DC) has been gaining momentum through 3GPP standardizing activities and industrial practices [4] [5]. The DC enables an MU to simultaneously communicate with a macro-cell base station (BS) and a small-cell access point (AP) via two different radio interfaces. With DC, the MUs can flexibly schedule traffic between the BS and AP, e.g., simultaneously sending its delay-sensitive small-volume data traffic to the BS and offloading delay-tolerant large-volume traffic to the AP. The new possibility offered by DC requires a newly designed data offloading scheme [6]–[8], [10]. In this paper, we consider the MUs' uplink traffic offloading problem with DC, and formulate the joint optimization of the BS' bandwidth allocation as well as the MUs' traffic scheduling and power allocation.

With the DC-capability, the MUs can flexibly schedule their traffic towards the macro and small cells, which yields different resource consumptions at the BS-side and the MU-side. For instance, if the MUs aggressively offload most of their traffic demand to small cell, this will reduce the bandwidth utilization at the BS-side, but at the expense of an increased transmit-power consumption of the MUs to combat the severe co-channel interferences when offloading traffic to the small cell. Hence there exists a tradeoff between the BS' bandwidth usage and the MUs' total transmit-power consumption. This motivates our consideration of the joint minimization of the BS' channel bandwidth consumption and the MUs' total transmit-power consumption.

- *From the MU's perspective*, to save its power consumption, the MU needs to carefully schedule its transmit-power and traffic towards the BS and AP. The co-channel interferences among MUs who offload data to the same AP complicate the MUs' joint traffic scheduling and power allocation problem. Specifically, if the MUs aggressively offload traffic to the AP, they may end up consuming significant more transmit-powers due to severe co-channel interferences.
- *From the BS' perspective*, due to valuable spectrum resources, the BS has the tendency of reducing the cellular spectrum allocation to the MUs and encourages the MUs to offload traffic to the AP. This may increase the MUs' power consumption as we explained before.

To fully exploit the benefits of data offloading via DC and properly control the MUs' co-channel interferences, we need to solve a joint optimization problem that considers the BS' bandwidth allocation to the MUs as well as the MUs' consequent traffic scheduling and power allocation. However, the non-convexity of this two-folded joint optimization problem makes it very difficult to solve. To efficiently solve this problem and compute the optimal offloading solution, we need to carefully explore the decomposable structure and unique properties of the joint optimization problem.

This paper's key contributions can be summarized as follows.

- *A Novel Traffic Offloading Scheme via DC*: We consider a new MU-uplink traffic offloading scenario with DC, which involves a joint optimization of the BS' bandwidth allocation to the MUs as well as the MUs' traffic scheduling and power allocation. We consider the MUs' co-channel interferences when offloading data to the AP, and aim to minimize the overall system cost

Y. Wu, Y. He, and L. Qian are with College of Information Engineering, Zhejiang University of Technology, Hangzhou, China, (email: iewuy@zjut.edu.cn, lpqian@zjut.edu.cn). L. Qian is the correspondence author.

J. Huang is with the Network Communications and Economics Lab, Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong (email: jwhuang@ie.cuhk.edu.hk).

X. Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (email: xshen@bbcr.uwaterloo.ca).

including both the BS' bandwidth usage and the MUs' total power consumption.

- *An Efficient Algorithm to Compute Optimal Offloading Solution*: By exploiting the decomposable structure of the problem, we propose an efficient algorithm to solve the non-convex joint optimization problem. The decomposition leads to a top-problem that optimizes the BS' bandwidth allocation and a consequent subproblem that optimizes the MUs' traffic scheduling and power allocation (under a given BS' bandwidth allocation). We then effectively solve the top-problem and subproblem by exploring their hidden monotonicity and convexity. Our proposed algorithm is able to effectively compute the optimal offloading solution.
- *Performance Gain of the Proposed Traffic Offloading Scheme:* Numerical results validate that our algorithm can compute the optimal offloading solution more than 90% faster than the standard LINGO solver. Our proposed traffic offloading scheme can also significantly reduce the system cost, namely, saving more than 60% of the cost compared with a fixed bandwidth allocation scheme and more than 75% of the cost compared with a fixed traffic scheduling scheme.

A. Literature Review

Since the seminal studies [21] [22], there have been many studies about optimal resource allocation for data offloading in cellular networks. In the following, we review two groups of studies close to our paper, namely, those about traffic offloading via DC and those about traffic offloading considering co-channel interference.

1) Traffic offloading with DC: In [6], Jha *et al.* provided a brief survey about the key technical challenges for small-cell DC. In [7], Mukherjee *et al.* proposed a pairing scheme that facilitates macro-cell BSs and small-cell APs to form DCs with MUs for traffic offloading. Considering the MU's limited transmit-power capacity to execute DC, Liu *et al.* proposed an enhanced uplink power control scheme that splits the MU's transmit-power budget to the macro-cell and small-cell in [8]. The issue of secrecy-provisioning in the DC-enabled traffic offloading over unlicensed band has been investigated in [9]. In [10] and [11], Mukherjee and Wang *et al.* studied flow control schemes for traffic offloading via DC, while taking into account the backhaul capacity constraint. In [12], Singh *et. al.* investigated a scenario where users' traffic is split across macro and small cells connected by non-ideal backhaul links, and developed an optimal traffic-splitting solution that accounts for the backhaul delay. To exploit the benefits of millimeter-wave, in [13], Semiari *et. al.* proposed a novel context-aware scheduling framework for dual-mode small base stations operating at millimeter-wave and microwave bands, with the objective of providing delay guarantees per user application. In [14], facilitated by the emerging DC-capability, Zakrzewska *et. al.* provided an overview of the network architecture with split control-plane and user-plane and proposed a scheme to implement the DC in LTE-Advanced heterogenous networks. Compared with these studies, the main novelty of our proposed traffic offloading scheme is that it involves a two-folded optimization problem including the BS' bandwidth allocation as well as the MUs' traffic scheduling and power allocation.

2) Traffic offloading considering co-channel interference: An important issue regarding traffic offloading is how to properly control co-channel interferences among the MUs when offloading traffic to the same AP. Related studies about this issue can be categorized into the following two subgroups.

- *Studies considering the interference in downlink data offloading:* In [23], Zhang *et al.* considered a case of downlink interference and proposed a power allocation scheme for access providers (which share a common spectrum) to enhance their downlink offloading capacities. In [24], Wang *et al.* exploited the MUs' device-to-device cooperation for data offloading and considered the MUs' interferences due to sharing the same channel. In [25], Ye *et al.* proposed a scheme for scheduling the MUs' traffic to different small-cells, while taking into account the inter-cell interference. In [26], Ho *et al.* investigated the load-coupling effect due to inter-cell interference and proposed a traffic allocation to distribute traffic between macro-cells and small-cell. The DC-enabled traffic offloading based on the emerging non-orthogonal multiple access has been investigated in [27]. In [28], Chen *et al.* investigated the time-varying traffic and proposed a dynamic decision model to compute the strategy for offloading traffic from macro-cells to small-cells. In [29], Iosifidis *et al.* proposed an auction mechanism that facilitates an efficient data offloading from mobile network operators to small-cell APs while taking into account the interferences among the APs.
- *Studies considering the interference in uplink data offloading:* There also exist studies considering the co-channel interferences in the MUs' uplink traffic offloading. In [16], Yang *et al.* considered the MUs' co-channel interferences when the MUs are offloaded to the AP and designed a refunding scheme to incentivize the privately-owned APs to admit the offloaded MUs. In [17], Kang *et al.* accounted for the MUs' co-channel interferences when offloading traffic to the AP, and designed access-selection schemes for maximizing the utility of network operator.

Our study in this paper takes into account the uplink co-channel interference when the MUs offload data to the AP, and hence is close to the second subgroup above. Compared with [16], [17], our proposed traffic offloading scheme incorporates the MUs' flexible traffic scheduling offered by DC, which leads to a two-folded optimization of the BS' bandwidth allocation to the MUs as well as the MUs' traffic scheduling and power allocation. Notice that without DC, each MU can only choose either BS or small cell to send its entire traffic demand (instead of flexibly scheduling traffic between macro and small cells), which may lead to significantly less efficient resource utilization. For instance, when many MUs offload their entire traffic demands to the small cell, the resulting severe channel interference will lead to a significant power consumption of the MUs. Despite the advantage of DC, the coupling between the BS' bandwidth allocation and the MUs' traffic scheduling and power allocation makes our problem much more challenging to solve.

II. System Model and Problem Decomposition

A. System Model and Problem Formulation

As illustrated in Figure 1, we consider a single operator who owns one macro-cell BS and serves a set $\mathcal{I} = \{1, 2, ..., I\}$ of MUs. These MUs are also under the coverage of a small cell (e.g., femtocell) AP owned by the same operator. Each MU i has a fixed uplink traffic demand R_i^{req}, and can simultaneously send its traffic to the BS and AP via two different radio interfaces[1]. Let r_{iB} denote MU i's traffic rate to the BS (the subscript "B" means "base station"), and r_{iA} denote MU i's offloaded traffic rate to the AP (the subscript "A" means "access point"). Let p_{iB} and p_{iA} denote MU i's transmit-powers to the BS and AP, respectively.

[1] In future cellular networks, the deployment of small cells will be more dense and homogeneous. Such a setting of dense small cells enables the MUs to easily find several suitable small cells to offload traffic, which thus facilitates the realization of the DC model studied in this work.

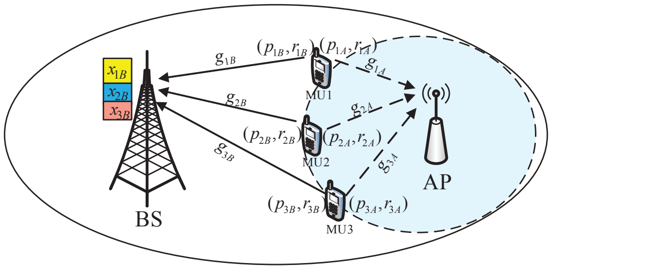


Fig. 1: An illustrative model with one BS, one AP, and three MUs.

The BS allocates orthogonal bands to different MUs' uplink transmissions, as in today's 4G cellular systems [15] [18]. Let x_{iB} denote the BS' bandwidth allocation for MU i. Hence MU i's uplink throughput to the BS is

$$r_{iB} = x_{iB} \log_2 \big(1 + \frac{p_{iB} g_{iB}}{x_{iB} n_0}\big), \forall i \in \mathcal{I}, \quad (1)$$

where g_{iB} denotes the channel power gain from MU i to the BS, and n_0 is the background noise density at the BS.

For the MUs' offloading to the same AP, we assume that they share the channel (e.g., [16], [17], [23]) and hence generate mutual interferences. Let W_A denote the AP's fixed bandwidth. The offloading-rate from MU i to the AP is given by

$$r_{iA} = W_A \log_2 \big(1 + \frac{p_{iA} g_{iA}}{\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA} + W_A n_0}\big), \forall i \in \mathcal{I}, \quad (2)$$

where g_{iA} denotes the channel power gain from MU i to the AP. The term $\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA}$ is the interference from other MUs to MU i. Although there are other transmission models (e.g., carrier sensing multiple access, CSMA) to quantify the offloading-rate [30], [31], the CSMA model is widely used for WiFi-based offloading and is used to quantify the user's successful packet-delivery rate by accounting for the collisions due to multiple users' overlapping packet-deliveries. In this work, instead of targeting for the WiFi-protocol and using the successful packet-delivery rate, we consider the general traffic offloading scenario through a small cell (e.g., femtocell) and quantify the MUs' achievable throughput from the information-theoretic perspective. Specifically, we consider the MUs' co-channel interferences when many MUs offload traffic to a small cell, and adopt the offloading throughput model in eq. (2) which is based on the Shannon's channel capacity. The model considers that each MU's transmission is interference-tolerable, and the small cell can provide each MU the offloading throughput according to the received signal to interference plus noise ratio (SINR). A similar model has been used in several related studies, e.g., [16]–[18].

We will consider the operator's optimization problem, which involves the minimization of a system cost function including both the MUs' total power consumption and the BS' bandwidth usage. We formulate the following JOINT optimization problem that jointly optimizes the BS' bandwidth allocation $\{x_{iB}\}_{i \in \mathcal{I}}$, as well as the MUs' traffic scheduling $\{r_{iA}, r_{iB}\}_{i \in \mathcal{I}}$ and transmit-powers $\{p_{iA}, p_{iB}\}_{i \in \mathcal{I}}$.

$$\underline{\text{(JOINT)}}: \quad \min \alpha \sum_{i \in \mathcal{I}} x_{iB} + (1-\alpha) \sum_{i \in \mathcal{I}} (p_{iA} + p_{iB}) \quad (3)$$

$$\begin{aligned} &\text{subject to: Constraints (1) and (2),} \\ &\quad r_{iB} + r_{iA} \geq R_i^{\text{req}}, \forall i \in \mathcal{I}, && (4) \\ &\quad 0 \leq p_{iA} \leq P_{iA}^{\max}, \forall i \in \mathcal{I}, && (5) \\ &\quad 0 \leq p_{iB} \leq P_{iB}^{\max}, \forall i \in \mathcal{I}, && (6) \\ &\quad x_B^{\min} \leq x_{iB} \leq x_B^{\max}, \forall i \in \mathcal{I}, && (7) \\ &\text{variables: } x_{iB},\ (r_{iA}, r_{iB}),\ \text{and } (p_{iA}, p_{iB}), \forall i \in \mathcal{I}. \end{aligned}$$

In (3), parameter α denotes the weight for the BS' bandwidth usage, and $(1-\alpha)$ denotes the weight for the MUs' total power consumption[2]. Constraint (4) ensures that MU i's traffic scheduling (r_{iA}, r_{iB}) meets its demand R_i^{req}. Constraints (5) and (6) ensure that MU i's transmit-powers to the AP and BS do not exceed the respective upper-bounds $P_{iA}^{\max}$ and $P_{iB}^{\max}$. In constraint (7), $x_B^{\min}$ denotes the BS' minimum bandwidth allocation for MU i (e.g., for signalling and hand-shaking with MU i). Meanwhile, $x_B^{\max}$ denotes the BS' maximum bandwidth allocation for MU i. For instance, in 3GPP LTE, the bandwidth allocation for each MU is scalable from 1.4MHz to 20MHz, and thus it is reasonable to impose $x_B^{\max}$ regarding the maximum bandwidth allocation for each MU. For notation compactness, we also denote the optimization variables in vector formats as $\boldsymbol{x}_B = \{x_{iB}\}_{i \in \mathcal{I}}$, $\boldsymbol{r}_A = \{r_{iA}\}_{i \in \mathcal{I}}$, $\boldsymbol{r}_B = \{r_{iB}\}_{i \in \mathcal{I}}$, $\boldsymbol{p}_A = \{p_{iA}\}_{i \in \mathcal{I}}$, and $\boldsymbol{p}_B = \{p_{iB}\}_{i \in \mathcal{I}}$. In particular, based on the recent 3GPP specification about DC [34] [35], we assume that the macro and small cells belong to a same operator, and they collaborate to accommodate the MUs' traffic via DC with the objective of minimizing the total resource consumption. An interesting future work is to further consider that macro and small cells belong to different network operators, and they selfishly optimize their own interests when accommodating the MUs' traffic via DC.

To avoid the trivial case that Problem (JOINT) is infeasible, we assume that the following is true in this paper

$$R_i^{\text{req}} \leq x_B^{\max} \log_2 (1 + \frac{p_{iB}^{\max} g_{iB}}{x_B^{\max} n_0}), \forall i \in \mathcal{I}. \quad (8)$$

B. Decomposition of Problem (JOINT) and Roadmap for Solution

There are two coupling effects in Problem (JOINT), namely, i) at the AP-side, the MUs' offloading-rates to the AP are coupled due to the non-convex constraint (2), and ii) at each MU-side: MU i's traffic scheduling decisions to the BS and AP are coupled due to (4). Hence Problem (JOINT) is a complicated non-convex optimization problem, which is difficult to solve in general. The key idea to solve Problem (JOINT) is to equivalently decompose it into a top-problem and a subproblem. Specifically, *the subproblem* is to optimize the MUs' traffic scheduling $(\boldsymbol{r}_A, \boldsymbol{r}_B)$ and transmit-powers $(\boldsymbol{p}_A, \boldsymbol{p}_B)$ under a given BS' bandwidth allocation $\boldsymbol{x}_B$. *The top-problem* is to further optimize $\boldsymbol{x}_B$ based on the optimal solution of the subproblem. Figure 3(a) at the end of Section III shows the connections between the top-problem and subproblem, and their details are as follows.

1) Subproblem to optimize $(\boldsymbol{r}_A, \boldsymbol{r}_B)$ and $(\boldsymbol{p}_A, \boldsymbol{p}_B)$ under a given $\boldsymbol{x}_B$: The subproblem (TPA) to optimize the MUs' Traffic scheduling and Power Allocation under a given $\boldsymbol{x}_B$ is given as

$$\begin{aligned} \underline{\text{(TPA)}}: \quad & V(\boldsymbol{x}_B) = \min \sum_{i \in \mathcal{I}} (p_{iA} + p_{iB}) \\ & \text{subject to: Constraints (1), (2), (4), (5), and (6),} \\ & \text{variables: } (\boldsymbol{r}_A, \boldsymbol{r}_B), \text{ and } (\boldsymbol{p}_A, \boldsymbol{p}_B). \end{aligned}$$

Notice that in subproblem (TPA), $\boldsymbol{x}_B$ is given in constraint (1). We use function $V(\boldsymbol{x}_B)$ to denote the optimal objective value of subproblem (TPA), as it depends on the given $\boldsymbol{x}_B$.

[2] A larger α means that more emphasis will be put on reducing the macro-cell BS' bandwidth usage, which thus leads to a smaller BS' bandwidth usage but with the cost of consuming a larger MUs' total transmit-power. By varying the value of α, we can characterize the Pareto-optimal operation points of the joint optimization problem. Figure 10 in Section V illustrates the impact of α on the optimal solution. Nevertheless, choosing the "best" (or the "right") value of α is very network-specific, and it depends on other constraints/considerations that need to be analytically characterized, which we consider as an interesting future direction for investigation.

2) Top-problem to optimize $\boldsymbol{x}_B$: After solving subproblem (TPA) and obtaining $V(\boldsymbol{x}_B)$, the top-problem (BA) to optimize the BS' Bandwidth Allocation is given by:

$$\underline{\text{(BA)}}: \quad \min \alpha \sum_{i \in \mathcal{I}} x_{iB} + (1-\alpha) V(\boldsymbol{x}_B)$$
$$\text{subject to: Constraint } (7),$$
$$\text{variables: } \boldsymbol{x}_B.$$

In the objective function of top-problem (BA), $V(\boldsymbol{x}_B)$ denotes the MUs' minimum total power consumption in the subproblem, given the BS' bandwidth allocation $\boldsymbol{x}_B$.

Based on the above decomposition structure, our key idea to solve Problem (JOINT) is to propose an efficient search algorithm over $\boldsymbol{x}_B$ subject to (7) (to solve the top-problem (BA)). In the search algorithm, for each being considered $\boldsymbol{x}_B$, we compute $V(\boldsymbol{x}_B)$ by solving the subproblem (TPA). Specifically, let $\boldsymbol{x}_B^* = \{x_{iB}^*\}_{i \in \mathcal{I}}$ denote the optimal solution of top-problem (BA). Moreover, let $(\boldsymbol{r}^*_{A,(\boldsymbol{x}_B^*)}, \boldsymbol{r}^*_{B,(\boldsymbol{x}_B^*)})$ and $(\boldsymbol{p}^*_{A,(\boldsymbol{x}_B^*)}, \boldsymbol{p}^*_{B,(\boldsymbol{x}_B^*)})$ denote the optimal solution of subproblem (TPA) under $\boldsymbol{x}_B^*$. We have the following result.

Proposition 1: *(Connection between Top-problem (BA) and Problem (JOINT))* The values of $\boldsymbol{x}_B^*$, $(\boldsymbol{r}^*_{A,(\boldsymbol{x}_B^*)}, \boldsymbol{r}^*_{B,(\boldsymbol{x}_B^*)})$, and $(\boldsymbol{p}^*_{A,(\boldsymbol{x}_B^*)}, \boldsymbol{p}^*_{B,(\boldsymbol{x}_B^*)})$ form the optimal solution of Problem (JOINT).

Proof: First, given the BS' bandwidth allocation $\boldsymbol{x}_B$, we notice that the resulting MUs' minimum total power consumption of subproblem (TPA), which is the optimal objective value $V(\boldsymbol{x}_B)$ of subproblem (TPA), is always unique. *Second*, top-problem (BA) aims to search for all possible $\boldsymbol{x}_B$ subject to (7) to minimize the system cost function $(1-\alpha)V(\boldsymbol{x}_B) + \alpha \sum_{i \in \mathcal{I}} x_{iB}$. Since $\boldsymbol{x}_B^*$ is the optimal solution of top-problem (BA), it yields the minimum value $(1-\alpha)V(\boldsymbol{x}_B^*) + \alpha \sum_{i \in \mathcal{I}} x_{iB}^*$, which corresponds to the minimum objective value of Problem (JOINT). In this sense, $\boldsymbol{x}_B^*$ is the optimal BS' bandwidth allocation of Problem (JOINT). Accordingly, $(\boldsymbol{r}^*_{A,(\boldsymbol{x}_B^*)}, \boldsymbol{r}^*_{B,(\boldsymbol{x}_B^*)})$ and $(\boldsymbol{p}^*_{A,(\boldsymbol{x}_B^*)}, \boldsymbol{p}^*_{B,(\boldsymbol{x}_B^*)})$, which are the optimal solutions of subproblem (TPA) under $\boldsymbol{x}_B^*$, are part of the optimal MUs' traffic scheduling and power allocation of Problem (JOINT), since they together yield $V(\boldsymbol{x}_B^*)$. ■

The key advantage of the above decomposition is that we can exploit the special properties of subproblem (TPA) and top-problem (BA) to solve them separately and efficiently. We also notice that the second-order cone programming (SOCP) is a useful tool to solve the non-convex optimization problem due to interference channel [32], [33]. However, the considerations of DC and the MUs' given traffic demands lead to Problem (JOINT) which is a non-convex two-sided resource allocation problem. Furthermore, the need to allocate BS' bandwidth to minimize the overall resource usage further complicates Problem (JOINT). Hence we need to exploit the special features of our problem to design an efficient solution methodology (which is one of the key contributions of our work).

III. Solving Subproblem (TPA)

In this section, we focus on solving subproblem (TPA) under a given $\boldsymbol{x}_B$ and evaluating $V(\boldsymbol{x}_B)$. We first turn subproblem (TPA) into an equivalent optimization problem through a series of transformations in Section III-A. We then explore the hidden convexity of the equivalent problem by further decomposing it into two subproblems in Section III-B. We propose efficient algorithms to solve the two subproblems in Sections III-C and III-D, respectively, which thus complete solving subproblem (TPA). Figure 2 at the end of this section summarizes the problem transformations and decomposition used in this section.

A. A Series of Equivalent Transformations on Subproblem (TPA)

The following property will help us understand the problem transformations.

Lemma *1: (Strict Bindingness of Traffic Demand Constraint)* Constraint (4) is always strictly binding at any optimal solution of subproblem (TPA).

Proof: The result can be proved by showing contradiction. We skip the details due to the limited space. ■

Lemma 1 enables us to express each MU i's (r_{iB}, p_{iB}) as functions of $\boldsymbol{p}_A$ as follows:

$$r_{iB} = R_i^{\text{req}} - W_A \log_2 \big(1 + \frac{p_{iA} g_{iA}}{\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA} + W_A n_0}\big). \tag{9}$$

$$p_{iB} = \frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \frac{1}{(1 + \frac{p_{iA} g_{iA}}{\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA} + W_A n_0})^{\frac{W_A}{x_{iB}}}} - \frac{x_{iB} n_0}{g_{iB}}. \tag{10}$$

Using (9) and (10) as well as (2), we can equivalently transform subproblem (TPA) into the following Power Allocation (PA) problem, which only uses $\boldsymbol{p}_A$ as a decision vector:

$$\underline{\text{(PA)}}: V(\boldsymbol{x}_B) = \min \sum_{i \in \mathcal{I}} \big(p_{iA} + \frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \frac{1}{(1 + \frac{p_{iA} g_{iA}}{\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA} + W_A n_0})^{\frac{W_A}{x_{iB}}}} - \frac{x_{iB} n_0}{g_{iB}}\big)$$

subject to:

$$W_A \log_2 \big(1 + \frac{p_{iA} g_{iA}}{\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA} + W_A n_0}\big) \leq R_i^{\text{req}}, \forall i \in \mathcal{I}, \tag{11}$$

$$p_{iA} \leq P_{iA}^{\max}, \forall i \in \mathcal{I}, \tag{12}$$

$$\frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \frac{1}{(1 + \frac{p_{iA} g_{iA}}{\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA} + W_A n_0})^{\frac{W_A}{x_{iB}}}} \leq P_{iB}^{\max} + \frac{x_{iB} n_0}{g_{iB}}, \forall i \in \mathcal{I}, \tag{13}$$

variables: $\boldsymbol{p}_A$.

However, Problem (PA) is still a complicated non-convex optimization. To further simplify Problem (PA), we introduce a new variable θ_{iA} as follows:

$$\theta_{iA} = \frac{p_{iA} g_{iA}}{\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA} + W_A n_0}, \forall i \in \mathcal{I}, \tag{14}$$

which represents MU i's achieved SINR at the AP. Let vector $\boldsymbol{\theta}_A = \{\theta_{iA}\}_{i \in \mathcal{I}}$. The following result shows the connection between $\boldsymbol{\theta}_A$ and $\boldsymbol{p}_A$.

Proposition 2: *(Connection between $\boldsymbol{\theta}_A$ and $\boldsymbol{p}_A$)* We have

$$p_{iA} = \frac{W_A n_0}{g_{iA}} \frac{\theta_{iA}}{1 + \theta_{iA}} \frac{1}{1 - \sum_{i \in \mathcal{I}} \frac{\theta_{iA}}{1 + \theta_{iA}}}, \forall i \in \mathcal{I}, \tag{15}$$

in which $\sum_{i \in \mathcal{I}} \frac{\theta_{iA}}{1+\theta_{iA}} < 1$ always holds.

Proof: We first prove that $\boldsymbol{p}_A$ can be uniquely determined by (15). We introduce a variable $z = \sum_{i \in \mathcal{I}} p_{iA} g_{iA} + W_A n_0$. Using z and (14), we obtain $\theta_{iA} = \frac{p_{iA} g_{iA}}{z - p_{iA} g_{iA}}, \forall i \in \mathcal{I}$, which further leads to $p_{iA} g_{iA} = z \frac{\theta_{iA}}{1+\theta_{iA}}, \forall i \in \mathcal{I}$. By summarizing both sides of this equation over all MUs in $\mathcal{I}$, we can derive $z = \frac{W_A n_0}{1 - \sum_{i \in \mathcal{I}} \frac{\theta_{iA}}{1+\theta_{iA}}}$, which leads to (15) because of $p_{iA} g_{iA} = z \frac{\theta_{iA}}{1+\theta_{iA}}, \forall i \in \mathcal{I}$. In particular, $\sum_{i \in \mathcal{I}} \frac{\theta_{iA}}{1+\theta_{iA}} < 1$ always holds in (15) as

$$\sum_{i \in \mathcal{I}} \frac{\theta_{iA}}{1 + \theta_{iA}} = \sum_{i \in \mathcal{I}} \frac{\frac{p_{iA} g_{iA}}{\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA} + W_A n_0}}{1 + \frac{p_{iA} g_{iA}}{\sum_{j \neq i, j \in \mathcal{I}} p_{jA} g_{jA} + W_A n_0}}$$
$$= \frac{\sum_{i \in \mathcal{I}} p_{iA} g_{iA}}{\sum_{i \in \mathcal{I}} p_{iA} g_{iA} + W_A n_0} < 1.$$

■

Eqs. (14) and (15) together provide a mapping between $\boldsymbol{p}_A$ and $\boldsymbol{\theta}_A$. Based on Proposition 2, we can use $\boldsymbol{\theta}_A$ to substitute $\boldsymbol{p}_A$ and equivalently transform Problem (PA) into the following SINR-Assignment (SINRA) problem which uses $\boldsymbol{\theta}_A$ as a decision vector:

<u>(SINRA)</u>:

$$V(\boldsymbol{x}_B) = \min \sum_{i\in\mathcal{I}} \Big(\frac{W_A n_0}{g_{iA}} \frac{\theta_{iA}}{1+\theta_{iA}} \frac{1}{1-\sum_{i\in\mathcal{I}} \frac{\theta_{iA}}{1+\theta_{iA}}} + \frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \frac{1}{(1+\theta_{iA})^{\frac{W_A}{x_{iB}}}} - \frac{x_{iB} n_0}{g_{iB}} \Big)$$

subject to: $0 \le \theta_{iA} \le 2^{\frac{R_i^{\text{req}}}{W_A}} - 1, \forall i \in \mathcal{I},$ (16)

$$\frac{W_A n_0}{g_{iA}} \frac{\theta_{iA}}{1+\theta_{iA}} \frac{1}{1-\sum_{i\in\mathcal{I}} \frac{\theta_{iA}}{1+\theta_{iA}}} \le P_{iA}^{\max}, \forall i \in \mathcal{I}, \tag{17}$$

$$\frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \frac{1}{(1+\theta_{iA})^{\frac{W_A}{x_{iB}}}} \le P_{iB}^{\max} + \frac{x_{iB} n_0}{g_{iB}}, \forall i \in \mathcal{I}, \tag{18}$$

$$\sum_{i\in\mathcal{I}} \frac{\theta_{iA}}{\theta_{iA}+1} < 1, \tag{19}$$

variables: $\boldsymbol{\theta}_A$.

To solve the non-convex optimization Problem (SINRA), we further introduce an auxiliary variable ρ_{iA} for each MU i and define a vector $\boldsymbol{\rho}_A = \{\rho_{iA}\}_{i\in\mathcal{I}}$. The mapping between $\boldsymbol{\rho}_A$ and $\boldsymbol{\theta}_A$ is

$$\rho_{iA} = \frac{\theta_{iA}}{1+\theta_{iA}}, \Longleftrightarrow \theta_{iA} = \frac{\rho_{iA}}{1-\rho_{iA}}, \forall i \in \mathcal{I}. \tag{20}$$

Using $\boldsymbol{\rho}_A$, we equivalently transform Problem (SINRA) into the following "Rho"-Assignment (RhoA) problem which only involves $\boldsymbol{\rho}_A$ as a decision vector:

<u>(RhoA)</u>: $$V(\boldsymbol{x}_B) = \min \sum_{i\in\mathcal{I}} \Big(\frac{W_A n_0}{g_{iA}} \frac{\rho_{iA}}{1-\sum_{i\in\mathcal{I}} \rho_{iA}} + \frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} (1-\rho_{iA})^{\frac{W_A}{x_{iB}}} - \frac{x_{iB} n_0}{g_{iB}} \Big)$$

subject to: $0 \le \rho_{iA} \le 1 - \frac{1}{2^{\frac{R_i^{\text{req}}}{W_A}}}, \forall i \in \mathcal{I},$ (21)

$$\frac{W_A n_0}{g_{iA}} \frac{\rho_{iA}}{1-\sum_{i\in\mathcal{I}} \rho_{iA}} \le P_{iA}^{\max}, \forall i \in \mathcal{I}, \tag{22}$$

$$1 - \rho_{iA} \le \left(\frac{P_{iB}^{\max} + \frac{x_{iB} n_0}{g_{iB}}}{\frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}}} \right)^{\frac{x_{iB}}{W_A}}, \forall i \in \mathcal{I}, \tag{23}$$

$$\sum_{i\in\mathcal{I}} \rho_{iA} < 1, \tag{24}$$

variables: $\boldsymbol{\rho}_A$.

Remark 1: (Equivalence between Problem (RhoA) and Subproblem (TPA)): Problem (RhoA) is equivalent to subproblem (TPA) for the following two reasons. *First*, based on the above transformations, the objective functions of Problem (RhoA) and subproblem (TPA) are same. *Second*, given $\boldsymbol{x}_B$, the feasible region of subproblem (TPA) is same as the feasible region given by (21)-(24). ■

Since Problem (RhoA) and subproblem (TPA) are equivalent, we will focus on solving Problem (RhoA) in the rest of this section. In particular, we will exploit the hidden convexity of Problem (RhoA) and propose an efficient algorithm to solve it.

B. Special Structure of Problem (RhoA) and its Decomposition

To solve Problem (RhoA), we introduce an additional nonnegative variable ρ_{0A}, whose purpose is to change constraint (24) into:

$$\sum_{i\in\mathcal{I}} \rho_{iA} + \rho_{0A} = 1.$$

By using ρ_{0A}, we can equivalently transform Problem (RhoA) into the following Problem (RhoA-E), which uses ρ_{0A} and $\boldsymbol{\rho}_A$ as decision variables (the letter "E" denotes "Equivalence").

<u>(RhoA-E)</u>: $$V(\boldsymbol{x}_B) = \min \sum_{i\in\mathcal{I}} \Big(\frac{W_A n_0}{g_{iA}} \frac{\rho_{iA}}{\rho_{0A}} + \frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} (1-\rho_{iA})^{\frac{W_A}{x_{iB}}} - \frac{x_{iB} n_0}{g_{iB}} \Big)$$

subject to: $0 \le \rho_{iA} \le 1 - \frac{1}{2^{\frac{R_i^{\text{req}}}{W_A}}}, \forall i \in \mathcal{I},$ (25)

$$\frac{W_A n_0}{g_{iA}} \frac{\rho_{iA}}{\rho_{0A}} \le P_{iA}^{\max}, \forall i \in \mathcal{I}, \tag{26}$$

$$1 - \rho_{iA} \le \left(\frac{P_{iB}^{\max} + \frac{x_{iB} n_0}{g_{iB}}}{\frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}}} \right)^{\frac{x_{iB}}{W_A}}, \forall i \in \mathcal{I}, \tag{27}$$

$$\sum_{i\in\mathcal{I}} \rho_{iA} + \rho_{0A} = 1, \tag{28}$$

variables: $\boldsymbol{\rho}_A$ and $\rho_{0A} > 0$.

Problem (RhoA-E) is equivalent to Problem (RhoA) via a simple variable-change. Let $\boldsymbol{\rho}^*_{A,(\boldsymbol{x}_B)}$ and $\rho^*_{0A,(\boldsymbol{x}_B)}$ denote the optimal solution of Problem (RhoA-E) under a given $\boldsymbol{x}_B$. Then, $\boldsymbol{\rho}^*_{A,(\boldsymbol{x}_B)}$ is the optimal solution of Problem (RhoA). Therefore, we aim at solving Problem (RhoA-E), which also solves Problem (RhoA) as well as subproblem (TPA) according to Remark 1.

The key to solve Problem (RhoA-E) efficiently is to exploit its hidden convexity. *Supposing that ρ_{0A} is fixed in advance, then the consequent subproblem of Problem (RhoA-E) is a convex optimization problem in $\boldsymbol{\rho}_A$, which is easy to solve [41].* Therefore, by enumerating $\rho_{0A} \in (0, 1]$ on the top level and solving the consequent series of subproblems for each given ρ_{0A}, we can solve Problem (RhoA-E). Based on this rationale, we present the top-problem to optimize $\rho_{0A} \in (0, 1]$ and the consequent subproblem to optimize $\boldsymbol{\rho}_A$ under a given ρ_{0A} as follows.

1) <u>The top-problem to optimize $\rho_{0A} \in (0, 1]$</u>: The top-problem to optimize $\rho_{0A} \in (0, 1]$ is given by

<u>(RhoA-E-Top)</u>: $$V(\boldsymbol{x}_B) = \min_{0<\rho_{0A}\le 1} V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A}),$$

where function $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A})$ is the optimal value of the subproblem (given below) under a given ρ_{0A}.

2) <u>The subproblem to optimize $\boldsymbol{\rho}_A$</u>: The sub-problem to optimize $\boldsymbol{\rho}_A$ under a given ρ_{0A} is given by

<u>(RhoA-E-Sub)</u>: $$V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A}) = \min \sum_{i\in\mathcal{I}} \Big(\frac{W_A n_0}{g_{iA}} \frac{\rho_{iA}}{\rho_{0A}} + \frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} (1-\rho_{iA})^{\frac{W_A}{x_{iB}}} - \frac{x_{iB} n_0}{g_{iB}} \Big)$$

subject to:

$$\rho_{iA}^{\text{lowest}} \le \rho_{iA} \le \min\left\{ 1 - \frac{1}{2^{\frac{R_i^{\text{req}}}{W_A}}}, P_{iA}^{\max} \rho_{0A} \frac{g_{iA}}{W_A n_0} \right\}, \forall i \in \mathcal{I}, \tag{29}$$

$$\sum_{i\in\mathcal{I}} \rho_{iA} = 1 - \rho_{0A}, \tag{30}$$

variables: $\boldsymbol{\rho}_A$.

Recall that the optimal value $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A})$ of Problem (RhoA-E-Sub) is used in Problem (RhoA-E-Top). In constraint (29), the lower-bound $\rho_{iA}^{\text{lowest}}$, which is derived from constraint (23), is given by

$$\rho_{iA}^{\text{lowest}} = \max\left\{0, 1 - \left(\frac{P_{iB}^{\max} + \frac{x_{iB}n_0}{g_{iB}}}{\frac{x_{iB}n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}}}\right)^{\frac{x_{iB}}{W_A}}\right\}, \forall i \in \mathcal{I}. \quad (31)$$

Remark 2: (Connection between Problem (RhoA-E) and Original Subproblem (TPA)) As explained before, Problem (RhoA-E) is equivalent to Problem (RhoA), and thus is equivalent to the original subproblem (TPA) in Section II-B. Therefore, in Problem (RhoA-E-Top), we denote the optimal value as $V(\boldsymbol{x}_B)$, i.e., the optimal value of subproblem (TPA) under a given BS' bandwidth allocation $\boldsymbol{x}_B$. Moreover, to differ from $V(\boldsymbol{x}_B)$, we use $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A})$ to denote the optimal value of Problem (RhoA-E-Sub). ■

As stated before, the advantage of decomposing Problem (RhoA-E) into Problem (RhoA-E-Sub) and Problem (RhoA-E-Top) is that Problem (RhoA-E-Sub) is a convex optimization problem.

C. Efficient Algorithm to Solve Problem (RhoA-E-Sub) under a Given ρ_{0A} and Compute $V^{Sub}_{(\boldsymbol{x}_B)}(\rho_{0A})$

We design an efficient algorithm to solve Problem (RhoA-E-Sub) in this subsection. Problem (RhoA-E-Sub) has the following important property.

Proposition 3: *(Convexity of Problem (RhoA-E-Sub))* Given $\rho_{0A} \in (0, 1]$, Problem (RhoA-E-Sub) is a convex optimization problem in $\boldsymbol{\rho}_A$.

Proof: Please refer to Appendix I. ■

Proposition 3 enables us to solve Problem (RhoA-E-Sub) based on the Karush-Kuhn-Tucker (KKT) conditions [41]. Specifically, in Problem (RhoA-E-Sub), we observe that only (30) couples the MUs' $\boldsymbol{\rho}_A$. Hence we introduce the dual-variable λ to relax (30) and obtain the following Lagrangian function:

$$\mathcal{L}_{(\rho_{0A})}(\boldsymbol{\rho}_A, \lambda) = \sum_{i\in\mathcal{I}} \left(\frac{W_A n_0}{g_{iA}} \frac{\rho_{iA}}{\rho_{0A}} + \frac{x_{iB}n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} (1-\rho_{iA})^{\frac{W_A}{x_{iB}}} - \frac{x_{iB}n_0}{g_{iB}} \right) + \lambda \left(1 - \rho_{0A} - \sum_{i\in\mathcal{I}} \rho_{iA} \right). \quad (32)$$

By taking the first-order derivative with respect to ρ_{iA}, we obtain:

$$\frac{\partial \mathcal{L}_{(\rho_{0A})}(\boldsymbol{\rho}_A, \lambda)}{\partial \rho_{iA}} = \frac{W_A n_0}{g_{iA}} \frac{1}{\rho_{0A}} - \lambda - \frac{x_{iB}n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \frac{W_A}{x_{iB}} (1-\rho_{iA})^{\frac{W_A}{x_{iB}}-1}. \quad (33)$$

With $\frac{\partial \mathcal{L}_{(\rho_{0A})}(\boldsymbol{\rho}_A, \lambda)}{\partial \rho_{iA}} = 0$, we can express ρ_{iA} as a function of λ as

$$\rho_{iA,(\rho_{0A})}(\lambda) = \left[1 - \left(\left(\frac{W_A n_0}{g_{iA}} \frac{1}{\rho_{0A}} - \lambda \right) \frac{1}{\frac{x_{iB}n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \frac{W_A}{x_{iB}}} \right)^{\frac{x_{iB}}{W_A - x_{iB}}} \right]_{\rho_{iA}^{\text{lowest}}}^{\rho_{iA,(\rho_{0A})}^{\text{upper}}}, \quad (34)$$

where $\rho_{iA}^{\text{lowest}}$ has been given in (31). Meanwhile, the upper-bound $\rho_{iA,(\rho_{0A})}^{\text{upper}}$, which depends on ρ_{0A} according to constraints (25) and (26), is given by

$$\rho_{iA,(\rho_{0A})}^{\text{upper}} = \min\left\{ 1 - \frac{1}{2^{\frac{R_i^{\text{req}}}{W_A}}}, \frac{g_{iA}}{W_A n_0} P_{iA}^{\max} \rho_{0A} \right\}, \forall i \in \mathcal{I}. \quad (35)$$

Eq. (34) shows that $\rho_{iA,(\rho_{0A})}(\lambda)$ is non-decreasing in λ. Thus, we can adopt the highly efficient bisection method [46] to determine the optimal multiplier $\lambda^*_{(\rho_{0A})}$ such that constraint (30) is binding. Then, by further substituting $\lambda^*_{(\rho_{0A})}$ into (34), the optimal solution for Problem (RhoA-E-Sub) can be given by $\rho^*_{iA,(\rho_{0A})} = \rho_{iA,(\rho_{0A})}(\lambda^*_{(\rho_{0A})})$. We thus finish solving Problem (RhoA-E-Sub).

DRhoA-Algorithm: to solve Problem (RhoA-E-Sub) and compute $\rho^*_{iA,(\rho_{0A})}$

```
1: Input: The BS' bandwidth allocation $\boldsymbol{x}_B$, and the AP's setting of $\rho_{0A}$.
2: The AP initializes gap$^{\text{Bis}}$ as a small number (e.g., gap$^{\text{Bis}} = 10^{-5}$) and broadcasts $\rho_{0A}$ to all MUs in $\mathcal{I}$.
3: Each MU $i$ calculates $\rho_{iA}^{\text{lowest}}$ (according to (31)) and $\rho_{iA,(\rho_{0A})}^{\text{upper}}$ (according to (35)).
4: Each MU $i$ calculates $H_{i,(\rho_{0A})}(\rho_{iA}^{\text{lowest}})$ and $H_{i,(\rho_{0A})}(\rho_{iA,(\rho_{0A})}^{\text{upper}})$ (according to (37)), and reports $\left(H_{i,(\rho_{0A})}(\rho_{iA}^{\text{lowest}}), H_{i,(\rho_{0A})}(\rho_{iA,(\rho_{0A})}^{\text{upper}})\right)$ to the AP.
5: The AP calculates $\lambda^{\text{lowest}}_{(\rho_{0A})}$ and $\lambda^{\text{upper}}_{(\rho_{0A})}$ (according to (36)), and sets $\underline{\lambda} = \lambda^{\text{lowest}}_{(\rho_{0A})}$ and $\overline{\lambda} = \lambda^{\text{upper}}_{(\rho_{0A})}$.
6: while $|\underline{\lambda} - \overline{\lambda}| >$ gap$^{\text{Bis}}$ do
7:     The AP sets $\lambda^{\text{cur}} = \frac{\underline{\lambda}+\overline{\lambda}}{2}$ and announces $\lambda^{\text{cur}}$ to all MUs.
8:     Each MU $i$ calculates $\rho_{iA,(\rho_{0A})}(\lambda^{\text{cur}})$ (according to (34)) and reports $\rho_{iA,(\rho_{0A})}(\lambda^{\text{cur}})$ to the AP.
9:     if The AP finds $\sum_{i\in\mathcal{I}} \rho_{iA,(\rho_{0A})}(\lambda^{\text{cur}}) > 1 - \rho_{0A}$ then
10:        The AP sets $\overline{\lambda} = \lambda^{\text{cur}}$.
11:    else
12:        The AP sets $\underline{\lambda} = \lambda^{\text{cur}}$.
13:    end if
14: end while
15: The AP sends a convergence-notification to all MUs in $\mathcal{I}$.
16: After receiving the convergence-notification, each MU $i$ calculates $p_{iA} = \frac{W_A n_0}{g_{iA}} \frac{\rho_{iA,(\rho_{0A})}(\lambda^{\text{cur}})}{\rho_{0A}}$ and $p_{iB} = \frac{x_{iB}n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \left(1 - \rho_{iA,(\rho_{0A})}(\lambda^{\text{cur}})\right)^{\frac{W_A}{x_{iB}}} - \frac{x_{iB}n_0}{g_{iB}}$, and reports $p_{iA}$ and $p_{iB}$ to the AP.
17: Output: The AP sets $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A}) = \sum_{i\in\mathcal{I}} (p_{iA} + p_{iB})$, and each MU $i$ sets $\rho^*_{iA,(\boldsymbol{x}_B,\rho_{0A})} = \rho_{iA,(\rho_{0A})}(\lambda^{\text{cur}})$.
```

Based on the above rationale, we propose a *Distributed "Rho" Assignment Algorithm* (i.e., DRhoA-Algorithm shown on Page 14) to solve Problem (RhoA-E-Sub) optimally. DRhoA-Algorithm, executed by the AP and MUs in a distributed manner, works as follows.

- The AP performs a bisection search over λ (in the while-loop from Line 6 to Line 14). Within each round of iteration, the AP announces the currently evaluated λ^{cur} to all MUs (Line 7). Then, each MU i determines its $\rho_{iA,(\rho_{0A})}(\lambda^{\text{cur}})$ (Line 8). Based on all MUs' feedbacks, the AP either updates its upper-bound $\overline{\lambda}$ (Line 10) or updates the lower-bound $\underline{\lambda}$ (Line 12) for λ.
- The above bisection search process continues until $\overline{\lambda}$ and $\underline{\lambda}$ are very close (by comparing with a small gap threshold gap$^{\text{Bis}}$ [3]), which ensures that λ^{cur} is equal to $\lambda^*_{(\rho_{0A})}$. Then, the AP sends a convergence-notification to all MUs (Line 15). Correspondingly, each MU i evaluates $\rho_{iA,(\rho_{0A})}(\lambda^{\text{cur}})$ and its current (p_{iA}, p_{iB}), and the AP outputs $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A})$ for Problem (RhoA-E-Sub) in Line 17.

Remark 3: (Distributed Nature of DRhoA-Algorithm) DRhoA-Algorithm is distributed by nature, since it does not require the AP to collect all MUs' private information, such as the channel gains $\{g_{iA}, g_{iB}\}_{i\in\mathcal{I}}$, the transmit-power capacities $\{P_{iA}^{\max}, P_{iB}^{\max}\}_{i\in\mathcal{I}}$, and the traffic demands $\{R_i^{\text{req}}\}_{i\in\mathcal{I}}$. Instead, each MU i, based on the AP's announced λ^{cur}, determines its own $\rho_{iA,(\rho_{0A})}(\lambda^{\text{cur}})$ and its transmit-powers (p_{iA}, p_{iB}) until reaching convergence. Hence there is no need for a central controller to collect all parameters.

We characterize the convergence of DRhoA-Algorithm in the following proposition, assuming that Problem (RhoA-E-Sub) is

[3] Parameter gap$^{\text{Bis}}$ denotes the specified tolerance for computational error in our DRhoA-Algorithm. We set gap$^{\text{Bis}}$ extremely small (namely, gap$^{\text{Bis}} = 10^{-5}$), such that DRhoA-Algorithm is guaranteed to find the optimal solution of Problem (RhoA-E-Sub) with a negligible computational error (which is no greater than 0.1% as verified in Tables I-III in Section V).

feasible. We will discuss the feasibility of Problem (RhoA-E-Sub) shortly afterwards.

Proposition 4: *(Convergence of DRhoA-Algorithm)* Assume that Problem (RhoA-E-Sub) is feasible under the given ρ_{0A}. With a sufficiently small gap$^{\text{Bis}}$, DRhoA-Algorithm is guaranteed to converge to the optimal solution of Problem (RhoA-E-Sub) within $\log_2\left(\frac{\lambda^{\text{upper}}_{(\rho_{0A})}-\lambda^{\text{lowest}}_{(\rho_{0A})}}{\text{gap}^{\text{Bis}}}\right)$ iterations, where $\lambda^{\text{lowest}}_{(\rho_{0A})}$ and $\lambda^{\text{upper}}_{(\rho_{0A})}$ are

$$\begin{aligned} \lambda^{\text{lowest}}_{(\rho_{0A})} &= \min_{i\in\mathcal{I}} \left\{H_{i,(\rho_{0A})}(\rho^{\text{lowest}}_{iA})\right\}, \\ \lambda^{\text{upper}}_{(\rho_{0A})} &= \max_{i\in\mathcal{I}} \left\{H_{i,(\rho_{0A})}(\rho^{\text{upper}}_{iA(\rho_{0A})})\right\}. \end{aligned} \tag{36}$$

In (36), the auxiliary function $H_{i,(\rho_{0A})}(\rho_{iA})$ (associated with each MU i under a given ρ_{0A}) is given by

$$H_{i,(\rho_{0A})}(\rho_{iA}) = \frac{W_A n_0}{g_{iA}}\frac{1}{\rho_{0A}} - \frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \frac{W_A}{x_{iB}} (1-\rho_{iA})^{\frac{W_A}{x_{iB}}-1}. \tag{37}$$

Proof: Please refer to Appendix II. ■

Finally, we discuss the feasibility of Problem (RhoA-E-Sub). Constraints (29) and (30) lead to the following sufficient and necessary conditions for the feasibility of Problem (RhoA-E-Sub):

$$\begin{aligned} &\rho^{\text{lowest}}_{iA} \leq \rho^{\text{upper}}_{iA,(\rho_{0A})}, \forall i \in \mathcal{I}, \text{ and} \\ &\sum_{i\in\mathcal{I}} \rho^{\text{lowest}}_{iA} \leq 1-\rho_{0A} \leq \sum_{i\in\mathcal{I}} \rho^{\text{upper}}_{iA,(\rho_{0A})}. \end{aligned} \tag{38}$$

The two conditions in (38) will be used in our following proposed algorithm to solve Problem (RhoA-E-Top).

D. Algorithm to Solve Problem (RhoA-E-Top) and Compute $V(\boldsymbol{x}_B)$

We continue to solve Problem (RhoA-E-Top) in this subsection, after using DRhoA-Algorithm to solve Problem (RhoA-E-Sub) and evaluating $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A})$ for the given ρ_{0A}.

The difficulty in solving Problem (RhoA-E-Top) is that we cannot derive $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A})$ analytically, which prevents us from using a gradient-based scheme to solve it. Fortunately, we notice that *Problem (RhoA-E-Top) only involves a single decision variable ρ_{0A}, which is constrained within a fixed interval $(0, 1]$*. This property allows us to use a simple line-search approach (with a small enough step-size) to enumerate $\rho_{0A} \in (0, 1]$ and solve Problem (RhoA-E-Top) directly. Based on this rationale and by using DRhoA-Algorithm as a subroutine, we propose LS-DRhoA-Algorithm (the letters "LS" stand for "line-search") to solve Problem (RhoA-E-Top). The details of LS-DRhoA-Algorithm are shown on Page 16.

LS-DRhoA-Algorithm works as follows. The AP enumerates $\rho_{0A} \in (0, 1]$ with a step-size Δ (i.e., the For-Loop from Line 3 to Line 12). For each announced ρ_{0A} by the AP, each MU i determines its $\rho^{\text{lowest}}_{iA}$ and $\rho^{\text{upper}}_{iA,(\rho_{0A})}$ and reports them back to the AP. If the currently evaluated ρ_{0A} ensures that Problem (RhoA-E-Sub) is feasible (in Line 6), then the AP and MUs use DRhoA-Algorithm (as a subroutine) to obtain the value of $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A})$ (in Line 7). Based on the outcome of the subroutine, the AP updates its current best value V^{cbv}, and each MU i updates its current best solution ρ^{cbs}_{iA} (Line 9). After finishing enumerating $\rho_{0A} \in (0, 1]$, the AP outputs the optimal solution of Problem (RhoA-E-Top) (Line 13). *As described in Remark 2, the optimal value of Problem (RhoA-E-Top) is equal to $V(\boldsymbol{x}_B)$ of subproblem (TPA). Hence we finish solving subproblem (TPA).* Since $\boldsymbol{x}_B$ is given in subproblem (TPA), we explicitly include the subscript $\boldsymbol{x}_B$ in the output of LS-DRhoA-Algorithm, i.e., $\rho^*_{0A,(\boldsymbol{x}_B)}$ and $\boldsymbol{\rho}^*_{A,(\boldsymbol{x}_B)}$.

Notice that in Line 6 of LS-DRhoA-Algorithm, conditions in (38) are used to determine the feasibility of Problem (RhoA-E-Sub). This step is important, since it helps avoid the computational burden if the chosen ρ_{0A} makes Problem (RhoA-E-Sub) infeasible.

LS-DRhoA-Algorithm: to solve Problem (RhoA-E-Top) and compute $V(\boldsymbol{x}_B)$

```
1: Input: The BS' bandwidth allocation $\boldsymbol{x}_B$.
2: The AP initializes $\Delta$ as a sufficiently small yet positive number and sets $V^{\text{cbv}}$ as a very large number.
3: for $\rho_{0A} = \Delta : \Delta : 1$ do
4:     The AP announces $\rho_{0A}$ to all MUs in $\mathcal{I}$.
5:     Each MU $i$ calculates $\rho^{\text{lowest}}_{iA}$ (in (31)) and $\rho^{\text{upper}}_{iA,(\rho_{0A})}$ (in (35)), and reports $(\rho^{\text{lowest}}_{iA}, \rho^{\text{upper}}_{iA,(\rho_{0A})})$ to the AP.
6:     if The AP finds that $\rho^{\text{upper}}_{iA,(\rho_{0A})} \geq \rho^{\text{lowest}}_{iA}, \forall i \in \mathcal{I}$ and $\sum_{i\in\mathcal{I}} \rho^{\text{lowest}}_{iA} \leq 1-\rho_{0A} \leq \sum_{i\in\mathcal{I}} \rho^{\text{upper}}_{iA,(\rho_{0A})}$ then
7:         The AP uses DRhoA-Algorithm to obtain $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A})$, and each MU $i$ obtains $\rho^*_{iA,(\boldsymbol{x}_B,\rho_{0A})}$.
8:         if The AP finds $V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A}) < V^{\text{cbv}}$ then
9:             The AP updates $V^{\text{cbv}} = V^{\text{Sub}}_{(\boldsymbol{x}_B)}(\rho_{0A})$ and $\rho^{\text{cbs}}_{0A} = \rho_{0A}$, and each MU $i$ updates $\rho^{\text{cbs}}_{iA} = \rho^*_{iA,(\boldsymbol{x}_B,\rho_{0A})}$.
10:        end if
11:    end if
12: end for
13: Output: The AP sets $V(\boldsymbol{x}_B) = V^{\text{cbv}}$ and $\rho^*_{0A,(\boldsymbol{x}_B)} = \rho^{\text{cbs}}_{0A}$. The AP sends $\rho^*_{0A,(\boldsymbol{x}_B)}$ to each MU. Each MU $i$ sets its $\rho^*_{iA,(\boldsymbol{x}_B)} = \rho^{\text{cbs}}_{iA}$.
```

We have the following result regarding the optimality of LS-DRhoA-Algorithm.

Proposition 5: *(Asymptotic Optimality)* As the chosen step-size Δ approaches to zero, LS-DRhoA-Algorithm is guaranteed to achieve the global optimum of Problem (RhoA-E-Top) as well as Problem (RhoA-E).

Proof: As Δ approaches to zero, LS-DRhoA-Algorithm asymptotically enumerates all $\rho_{0A} \in (0, 1]$. According to Proposition 4, DRhoA-Algorithm enables the AP and all MUs to obtain the optimal solution of Problem (RhoA-E-Sub) for each given ρ_{0A}. Hence by using DRhoA-Algorithm as a subroutine, the solution found by LS-DRhoA-Algorithm will be the global optimum of Problem (RhoA-E-Top) and Problem (RhoA-E). ■

Based on Proposition 5 and the equivalence between Problem (RhoA-E) and original subproblem (TPA), we can solve subproblem (TPA) optimally. In particular, by using the output of LS-DRhoA-Algorithm (i.e., $\rho^*_{0A,(\boldsymbol{x}_B)}$ and $\rho^*_{iA,(\boldsymbol{x}_B)}$), each MU i can individually determine its optimal traffic scheduling and power allocation for subproblem (TPA). The details are as follows.

Proposition 6: *(Optimal Traffic Scheduling and Power Allocation of Subproblem (TPA)):* Based on the output of LS-DRhoA-Algorithm (i.e., $\rho^*_{iA,(\boldsymbol{x}_B)}$ and $\rho^*_{0A,(\boldsymbol{x}_B)}$), each MU i can individually determine its optimal traffic scheduling and power allocation for subproblem (TPA) as follows (the subscript $\boldsymbol{x}_B$ is included in all the following optimal solutions, since subproblem (TPA) is subject to the given $\boldsymbol{x}_B$):

$$r^*_{iA,(\boldsymbol{x}_B)} = -W_A \log_2\left(1-\rho^*_{iA,(\boldsymbol{x}_B)}\right), \tag{39}$$

$$p^*_{iA,(\boldsymbol{x}_B)} = \frac{W_A n_0}{g_{iA}} \frac{\rho^*_{iA,(\boldsymbol{x}_B)}}{\rho^*_{0A,(\boldsymbol{x}_B)}}, \tag{40}$$

$$r^*_{iB,(\boldsymbol{x}_B)} = R_i^{\text{req}} + W_A \log_2\left(1-\rho^*_{iA,(\boldsymbol{x}_B)}\right), \tag{41}$$

$$p^*_{iB,(\boldsymbol{x}_B)} = \frac{x_{iB} n_0}{g_{iB}} 2^{\frac{R_i^{\text{req}}}{x_{iB}}} \left(1-\rho^*_{iA,(\boldsymbol{x}_B)}\right)^{\frac{W_A}{x_{iB}}} - \frac{x_{iB} n_0}{g_{iB}}. \tag{42}$$

Proof: Given $\rho^*_{iA,(\boldsymbol{x}_B)}$ and $\rho^*_{0A,(\boldsymbol{x}_B)}$, each MU's optimal offloading-rate to the AP in (39) is obtained via (2), (14), and (20), and its corresponding transmit-power to the AP in (40) is obtained via (15) and (20). Meanwhile, each MU's traffic rate to the BS in (42) is obtained via (9), and its corresponding transmit-power to the BS in (42) is obtained via (10). ■

Based on Proposition 6, after executing LS-DRhoA-Algorithm, each MU can individually determine its optimal traffic scheduling and power allocation for subproblem (TPA). Moreover, by exploiting the distributed nature of DRhoA-Algorithm (i.e., Remark 3), LS-DRhoA-Algorithm requires a limited rounds of message exchanges between the MUs and AP. Thus, LS-DRhoA-Algorithm can be implemented in a distributed manner. In summary, the proposed LS-DRhoA-Algorithm can be easily implemented in practice.

Remark 4: (Total Number of Iterations used by LS-DRhoA-Algorithm) The total number of iterations used by LS-DRhoA-Algorithm can be calculated as follows. First, LS-DRhoA-Algorithm requires at most $\frac{1}{\Delta}$ rounds of iterations for enumerating $\rho_{0A} \in (0, 1]$. Second, for each enumerated ρ_{0A}, at most $\log_2\big(\frac{\lambda^{\text{upper}}_{(\rho_{0A})}-\lambda^{\text{lowest}}_{(\rho_{0A})}}{\text{gap}^{\text{Bis}}}\big)$ rounds of iterations are required to perform DRhoA-Algorithm according to Proposition 4 (recall that $\lambda^{\text{upper}}_{(\rho_{0A})}$ and $\lambda^{\text{lowest}}_{(\rho_{0A})}$ are given in (36)). In particular, notice that in each round of iteration in DRhoA-Algorithm, no iterative computation is required, since we have derived all required calculations in closed forms. In summary, LS-DRhoA-Algorithm requires no more than $\frac{1}{\Delta}\log_2\big(\frac{\lambda^{\text{upper}}_{(\rho_{0A})}-\lambda^{\text{lowest}}_{(\rho_{0A})}}{\text{gap}^{\text{Bis}}}\big)$ rounds of iterations. In Section V, we evaluate the performance (i.e., the accuracy and computational time) of LS-DRhoA-Algorithm by using different values of Δ. The numerical results show that using a moderately small step-size Δ in LS-DRhoA-Algorithm can yield a solution sufficiently close to the global optimum. ■

Until now, we solve subproblem (TPA) and compute the value of $V(\boldsymbol{x}_B)$ for a given $\boldsymbol{x}_B$. Figure 2(a) summarizes the transformations that transform subproblem (TPA) into Problems (RhoA-E-Top) and (RhoA-E-Sub). Figure 2(b) shows the connections between our LS-DRhoA-Algorithm and its subroutine DRhoA-Algorithm, which solve Problems (RhoA-E-Top) and (RhoA-E-Sub), respectively. Before leaving this section, we notice that subproblem (TPA) might be infeasible under some given $\boldsymbol{x}_B$. Thus, we propose an algorithm, which can also be implemented in a distributed manner, to check with the feasibility of subproblem (TPA) in Appendix III.

IV. Solving Top-problem (BA)

After computing $V(\boldsymbol{x}_B)$ for each given $\boldsymbol{x}_B$, we then solve top-problem (BA) to find the optimal BS' bandwidth allocation $\boldsymbol{x}_B^*$. We first identify the monotonicity of $V(\boldsymbol{x}_B)$, based on which we transform top-problem (BA) into an equivalent monotonic optimization problem and propose an algorithm to solve it.

A. Brief Introduction of Monotonic Optimization

We first briefly introduce the monotonic optimization in this subsection [42] [43]. We start with two important definitions.

Definition 1: (Normal Set). A set $\mathcal{G} \subset \mathcal{R}_+^n$ is normal, if for any two points $\boldsymbol{x}$ and $\boldsymbol{x}' \in \mathcal{R}_+^n$ with $\boldsymbol{x}' \leq \boldsymbol{x}$[4].

Definition 2: (Reverse Normal Set). A set $\mathcal{H} \subset \mathcal{R}_+^n$ is a reversed normal set, if for two points $\boldsymbol{x}$ and $\boldsymbol{x}' \in \mathcal{R}_+^n$ with $\boldsymbol{x}' \geq \boldsymbol{x}$ and $\boldsymbol{x} \in \mathcal{H}$, we always have $\boldsymbol{x}' \in \mathcal{H}$.

Based on the above definitions, a canonic form of the monotonic optimization problem is as follows.

$$\max_{\boldsymbol{x}} f(\boldsymbol{x}), \text{ subject to: } \boldsymbol{x} \in \mathcal{G} \cap \mathcal{H}, \tag{43}$$

where $f(\boldsymbol{x}) : \mathcal{R}_+^n \rightarrow \mathcal{R}$ is an increasing function[5]. Set $\mathcal{G} \subset [\boldsymbol{0}, \boldsymbol{b}]$ is a normal set with nonempty interior, and set $\mathcal{H}$ is a reverse normal set on $[\boldsymbol{0}, \boldsymbol{b}]$, with vector $\boldsymbol{b}$ representing a given point in $\mathcal{R}_+^n$.

As shown in (43), a monotonic optimization problem involves maximizing a monotonic objective function subject to a feasible region constructed by the intersection of a normal set and a reversed normal set [42] [43]. The monotonicity inherent in the problem enables us to design very efficient algorithm to solve the problem. Specifically, using the monotonicity of the constraints, one can iteratively construct a group of *poly-blocks* to approximate the feasible region with increasing precisions. Furthermore, thanks to the monotonicity of the objective function, the optimal solution is guaranteed to lie at one of the vertices of the constructed poly-blocks, as long as this vertex falls within (or very close enough to) the feasible region. Based on this idea, one can design the *poly-block outer-approximation algorithm*, which can efficiently search for the globally optimal solution of a monotonic optimization problem.

B. Monotonic of Top-problem (BA) and Algorithm Design

The difficulty in solving top-problem (BA) is that we cannot derive the analytical expression of $V(\boldsymbol{x}_B)$. The key idea to solve top-problem (BA) is to exploit its hidden monotonic property. To this end, we first show the following important property.

Proposition 7: The optimal value $V(\boldsymbol{x}_B)$ of subproblem (TPA) is decreasing in $\boldsymbol{x}_B$.

Proof: Please refer to Appendix IV. ■

Proposition 7 is consistent with the intuition. Given a larger BS' bandwidth allocation, the MUs' minimum total power consumption to achieve their required traffic demands always decreases.

Based on Proposition 7, we know that the objective function of top-problem (BA) is structured by the difference of two increasing functions as follows:

$$\alpha \sum_{i\in\mathcal{I}} x_{iB} + (1-\alpha)\, V(\boldsymbol{x}_B) = -\left((\alpha-1)\, V(\boldsymbol{x}_B) - \alpha \sum_{i\in\mathcal{I}} x_{iB}\right), \tag{44}$$

where function $(\alpha - 1)\, V(\boldsymbol{x}_B)$ is increasing in $\boldsymbol{x}_B$ (according to Proposition 7). In (44), we intentionally put a negative sign outside of the difference in order to transform top-problem (BA) into a monotonic optimization that maximizes an objective function. Such an operation will ease our following algorithmic design to compute the optimal BS' bandwidth allocation.

The property shown in (44) (i.e., the difference between two increasing functions) enables us to transform top-problem (BA) into a monotonic optimization problem [42] [43]. The details are as follows. We introduce an auxiliary variable ω such that:

$$\sum_{i\in\mathcal{I}} x_{iB} + \omega = \sum_{i\in\mathcal{I}} x_B^{\max}. \tag{45}$$

Therefore, the feasible interval of the introduced ω can be given by

$$0 \leq \omega \leq \sum_{i\in\mathcal{I}} x_B^{\max} - \sum_{i\in\mathcal{I}} x_B^{\min}. \tag{46}$$

Using ω, (44), (45), and (46), we equivalently transform top-problem (BA) into the following one:

$$\underline{\text{(BA-Monotonic)}}: \quad \max\, (\alpha-1)\, V(\boldsymbol{x}_B) + \alpha\left(\omega - \sum_{i\in\mathcal{I}} x_B^{\max}\right)$$
$$\text{subject to: } (\boldsymbol{x}_B, \omega) \in \mathcal{G} \cap \mathcal{H},$$
$$\text{variables: } \boldsymbol{x}_B \text{ and } \omega.$$

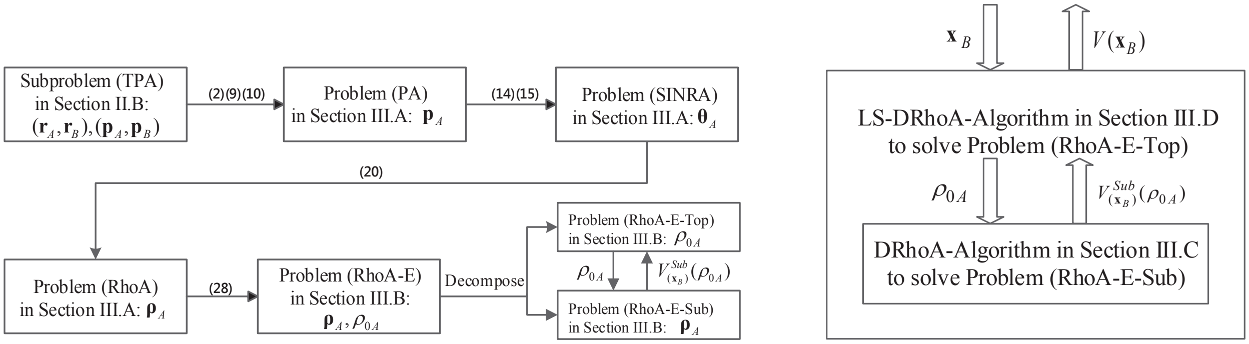
Sets $\mathcal{G}$ and $\mathcal{H}$ are respectively given by

$$\mathcal{G} = \big\{(\boldsymbol{x}_B, \omega) | 0 \leq x_{iB} \leq x_B^{\max}, 0 \leq \omega \leq \sum_{i\in\mathcal{I}} x_B^{\max} - \sum_{i\in\mathcal{I}} x_B^{\min}, \sum_{i\in\mathcal{I}} x_{iB} + \omega \leq \sum_{i\in\mathcal{I}} x_B^{\max}\big\}, \tag{47}$$

$$\mathcal{H} = \big\{(\boldsymbol{x}_B, \omega) | x_{iB} \geq x_B^{\min}, \omega \geq 0\big\}. \tag{48}$$

[4] We say that two points $\boldsymbol{x}$ and $\boldsymbol{x}' \in \mathcal{R}_+^n$ satisfy $\boldsymbol{x}' \leq \boldsymbol{x}$, if $x'_k \leq x_k$ holds for each element-index k of the two vectors. and $\boldsymbol{x} \in \mathcal{G}$, we always have $\boldsymbol{x}' \in \mathcal{G}$.

[5] Specifically, given two different $\boldsymbol{x}$ and $\boldsymbol{x}'$ with $x_k \geq x'_k, \forall k$ and $x_j > x'_j$ for at at least one index j, there always exists $f(\boldsymbol{x}) < f(\boldsymbol{x}')$.


Subproblem (TPA) in Section II.B: $(\mathbf{r}_A,\mathbf{r}_B),(\mathbf{p}_A,\mathbf{p}_B)$
(2)(9)(10)
Problem (PA) in Section III.A: $\mathbf{p}_A$
(14)(15)
Problem (SINRA) in Section III.A: $\boldsymbol{\theta}_A$
(20)
Problem (RhoA) in Section III.A: $\boldsymbol{\rho}_A$
(28)
Problem (RhoA-E) in Section III.B: $\boldsymbol{\rho}_A, \rho_{0A}$
Decompose
Problem (RhoA-E-Top) in Section III.B: ρ_{0A}
ρ_{0A}
$V^{Sub}_{(\mathbf{x}_B)}(\rho_{0A})$
Problem (RhoA-E-Sub) in Section III.B: $\boldsymbol{\rho}_A$
$\mathbf{x}_B$
$V(\mathbf{x}_B)$
LS-DRhoA-Algorithm in Section III.D to solve Problem (RhoA-E-Top)
ρ_{0A}
$V^{Sub}_{(\mathbf{x}_B)}(\rho_{0A})$
DRhoA-Algorithm in Section III.C to solve Problem (RhoA-E-Sub)

(a)

(b)

Fig. 2: Subfigure (a): A series of equivalent transformations that transform subproblem (TPA) to Problems (RhoA-E-Top) and (RhoA-E-Sub). Subfigure (b): Connections between our proposed LS-DRhoA-Algorithm and its subroutine DRhoA-Algorithm.

Based Definitions 1 and 2, we have the following result.

Lemma *2:* Set $\mathcal{G}$ given in (47) is a normal set, and set $\mathcal{H}$ given in (48) is a reversed normal set.

Proof: Both $\mathcal{G}$ and $\mathcal{H}$ are defined in $\mathcal{R}^n_+$. We next prove that $\mathcal{G}$ is a normal set. Suppose that there exist a point $(\boldsymbol{x}_B, \omega) \in \mathcal{G}$ and another point $(\boldsymbol{x}'_B, \omega')$ satisfying $(\boldsymbol{x}'_B, \omega') \leq (\boldsymbol{x}_B, \omega)$. Since $\sum_{i\in\mathcal{I}} x_{iB} + \omega$ is strictly increasing in $(\boldsymbol{x}_B, \omega)$, we know that point $(\boldsymbol{x}'_B, \omega') \in \mathcal{G}$ always holds. Thus, based on the definition of normal set (i.e., Definition 1), we conclude that set $\mathcal{G}$ is a normal set.

With a similar argument, we can prove that $\mathcal{H}$ is a reversed normal set. Suppose that there exist point $(\boldsymbol{x}_B, \omega) \in \mathcal{H}$ and point $(\boldsymbol{x}'_B, \omega')$ satisfying $(\boldsymbol{x}'_B, \omega') \geq (\boldsymbol{x}_B, \omega)$. It is easy to know that $(\boldsymbol{x}'_B, \omega') \in \mathcal{H}$ always holds. Based on the definition of reversed normal set (i.e., Definition 2), we conclude that set $\mathcal{H}$ is a reversed normal set. ■

Based on Lemma 2, we have the following important proposition regarding Problem (BA-Monotonic).

Proposition 8: *(Monotonicity of Problem (BA-Monotonic))* Problem (BA-Monotonic) is a monotonic problem in $(\boldsymbol{x}_B, \omega)$.

Proof: The objective function of Problem (BA-Monotonic) is increasing, and the feasible region is structured by the intersection of $\mathcal{G}$ (a normal set) and $\mathcal{H}$ (a reversed normal set). Following the monotonic optimization theory [42] [43], we can show that Problem (BA-Monotonic) is a monotonic optimization problem. ■

Proposition 8 enables us to adopt the poly-block outer-approximation algorithm to solve Problem (BA-Monotonic) and compute the optimal solution $\boldsymbol{x}^*_B$. The details of our algorithm are shown in the following Poly-Block approximation based Bandwidth Allocation Algorithm (i.e., PBBA-Algorithm). In our PBBA-Algorithm, a tuple of $(\boldsymbol{x}_B, \omega)$ corresponds to a vertex. In particular, given a vertex $(\boldsymbol{x}_B, \omega)$, our PBBA-Algorithm uses LS-DRhoA-Algorithm (proposed in Section III) as a subroutine to compute $V(\boldsymbol{x}_B)$. Our PBBA-Algorithm works as follows.

PBBA-Algorithm: to solve top-problem (BA) and compute $\boldsymbol{x}^*_B$

```
1: The BS initializes the current best solution $CBS = \emptyset$, the current best value $CBV = -\infty$, the iteration-index $k = 1$, and $\delta$ as a small positive number (e.g., $\delta = 0.001$). The BS also sets the flag for stopping as $f_{\text{stop}} = 0$, and initializes the vertex-set $\mathcal{T}_1$ to have a single vertex as $\mathcal{T}_1 = \{(\{x_B^{\max}\}_{i\in\mathcal{I}}, \sum_{i\in\mathcal{I}} x_B^{\max} - \sum_{i\in\mathcal{I}} x_B^{\min})\}$.
2: while $f_{\text{stop}} = 0$ do
3:    The BS selects the current best vertex $\boldsymbol{z}^k \in \arg\max\{C(\boldsymbol{z})|\boldsymbol{z} \in \mathcal{T}_k\}$. Specifically, for each vertex $\boldsymbol{z} = (\boldsymbol{x}_B, \omega) \in \mathcal{T}_k$, the BS instructs the AP and the MUs to perform LS-DRhoA-Algorithm such that each MU can obtain $(r^*_{iA,(\boldsymbol{x}_B)}, r^*_{iB,(\boldsymbol{x}_B)})$ and $(p^*_{iA,(\boldsymbol{x}_B)}, p^*_{iB,(\boldsymbol{x}_B)})$ according to Proposition 6. Each MU sends its $(p^*_{iA,(\boldsymbol{x}_B)}, p^*_{iB,(\boldsymbol{x}_B)})$ to the BS, and the BS can compute $V(\boldsymbol{x}_B) = \sum_{i\in\mathcal{I}}(p^*_{iA,(\boldsymbol{x}_B)} + p^*_{iB,(\boldsymbol{x}_B)})$ and $C(\boldsymbol{z}) = (\alpha - 1)V(\boldsymbol{x}_B) + \alpha(\omega - \sum_{i\in\mathcal{I}} x_B^{\max})$.
4:    The BS constructs a line between origin and $\boldsymbol{z}^k$, and finds the intersection point $\boldsymbol{y}^k$ between the above constructed line and the upper boundary given in $\mathcal{G}$ (via using bisection search). The BS computes $C(\boldsymbol{y}^k)$ by using the same procedure in Step 3.
5:    if $C(\boldsymbol{y}^k) > CBV$ then
6:       The BS updates $CBV = C(\boldsymbol{y}^k)$ and sets $CBS = \boldsymbol{y}^k$.
7:    end if
8:    if $\parallel \boldsymbol{y}^k - \boldsymbol{z}^k \parallel < \delta$ then
9:       The BS sets $f_{\text{stop}} = 1$.
10:   end if
11:   The BS updates the vertex-set as $\mathcal{T}_{k+1} = (\mathcal{T}_k \backslash \{\boldsymbol{z}^k\}) \cup \{\boldsymbol{z}^k + (y^k_j - z^k_j)\boldsymbol{e}_j\}$. The BS then remove all vertexes $\boldsymbol{z} \in \mathcal{T}_{k+1} \backslash \mathcal{H}$ (for clearing up those infeasible vertexes).
12:   if $\mathcal{T}_{k+1}$ is empty then
13:      The BS sets $f_{\text{stop}} = 1$.
14:   end if
15:   The BS sets $k = k + 1$.
16: end while
17: Output: The BS sets $\boldsymbol{x}^*_B$ according to vertex $CBS$.
```

The key component of PBBA-Algorithm is the While-Loop (Lines 2-16), whose purpose is to iteratively construct the poly-blocks that approximate the upper-boundary of $\mathcal{G} \cap \mathcal{H}$ (given by (47) and (48)) as much as possible. In the k-th iteration, set $\mathcal{T}_k$ denotes the current set of vertexes. In $\mathcal{T}_k$, the BS finds a best vertex $\boldsymbol{z}^k = (\boldsymbol{x}_B, \omega)$ that yields the largest objective value for Problem (BA-Monotonic) in Step 3 (for simplicity, we use $C(\boldsymbol{z}^k) = (\alpha - 1)V(\boldsymbol{x}_B) + \alpha(\omega - \sum_{i\in\mathcal{I}} x_B^{\max})$ to denote the objective function of Problem (BA-Monotonic)). With $\boldsymbol{z}^k$, the BS executes two tasks as follows:

- *Task i): to update the current best solution (CBS) and the current best value (CBV).* The BS first constructs a line from origin to $\boldsymbol{z}^k$. It then finds the intersection point (denoted by $\boldsymbol{y}^k$) between the constructed line and the upper-boundary of the feasible region (Line 4)[6]. The BS uses $C(\boldsymbol{y}^k)$ and $\boldsymbol{y}^k$ to update the CBV and the CBS in the k-th iteration (Lines 5-7).
- *Task ii): to construct poly-blocks $\mathcal{T}_{k+1}$ for the next round iteration.* The BS uses vertex $\boldsymbol{z}^k$ and the intersection $\boldsymbol{y}^k$ to construct the new poly-blocks that approximate $\mathcal{G} \cap \mathcal{H}$ with increasing precisions (Line 11)[7]. The purpose of Line 11 is to remove the region in which the optimal solution does not exist.

After the BS uses PBBA-Algorithm to compute the optimal bandwidth allocation $\boldsymbol{x}^*_B$ for all MUs, the BS can further notify the AP and all MUs to execute LS-DRhoA-Algorithm such that each MU i can obtain its $\rho^*_{iA,(\boldsymbol{x}^*_B)}$ and $\rho^*_{0A,(\boldsymbol{x}^*_B)}$. After that, each MU i can indi-

[6] Thanks to the monotonicity of the constraints, we can use the bisection method to find $\boldsymbol{y}^k$ very efficiently.

[7] In Line 11, scalar y^k_j (or z^k_j) denotes the j-th element of vector $\boldsymbol{y}^k$ (or vector $\boldsymbol{z}^k$). Vector $\boldsymbol{e}_j$ denotes a vector with the j-th element equal to 1, and all the other elements are equal to 0. In PBBA-Algorithm, all vectors are of dimension $1 \times (I+1)$, with I denoting the number of the MUs.

vidually compute its optimal traffic scheduling $\left(r^*_{iA,(\boldsymbol{x}^*_B)}, r^*_{iB,(\boldsymbol{x}^*_B)}\right)$ and optimal power allocation $\left(p^*_{iA,(\boldsymbol{x}^*_B)}, p^*_{iB,(\boldsymbol{x}^*_B)}\right)$ according to Proposition 6. *We thus finish solving the original Problem (JOINT).*

Figure 3(a) shows the connections between top-problem (BA) and Problem (BA-Monotonic) regarding the decomposition of original Problem (JOINT). Figure 3(b) shows the connections between our PBBA-Algorithm to solve Problem (BA-Monotonic) and the subroutine LS-DRhoA-Algorithm to solve subproblem (TPA).

Remark 5: (Distributed Nature of Proposed Methodology to Solve Problem (JOINT)) We emphasize that the proposed algorithm to solve the original Problem (JOINT) can be executed in a distributed manner. *First*, the operations of PBBA-Algorithm only require a limited rounds of message exchanges between the BS and the MUs. *Second*, as described close to the end of Section III-D, LS-DRhoA-Algorithm (i.e., the subroutine of PBBA-Algorithm) is ready to be implemented in a distributed manner. *Third*, in the final step to compute $\left(r^*_{iA,(\boldsymbol{x}^*_B)}, r^*_{iB,(\boldsymbol{x}^*_B)}\right)$ and $\left(p^*_{iA,(\boldsymbol{x}^*_B)}, p^*_{iB,(\boldsymbol{x}^*_B)}\right)$, each MU i only needs to know its own x^*_{iB} allocated by the BS, without requiring to know other MUs' $\{x^*_{jB}\}_{j\neq i}$. In summary, the whole proposed algorithm to solve the original Problem (JOINT) can be implemented in a distributed manner. ■

According to [42], [43], however, it is technically challenging to quantify the number of iterations required by the poly-block approximation method (which is still an open question in the field of optimization). Therefore, in Section V, we execute extensive numerical testings to validate the efficiency of our PBBA-Algorithm. The results demonstrate that our PBBA-Algorithm takes a very short computational time to find the optimal solution, i.e., 90% less than that required by LINGO[8].

V. Numerical Results

In this section, we numerically validate the proposed LS-DRhoA-Algorithm and PBBA-Algorithm, and show the performance of our proposed data offloading scheme. We consider the following simulation scenario.

Network topology and channel gains: The BS is located at the origin, and the AP is located at (350m,0m). The MUs are randomly located within a circle whose center is located at $(320\text{m}, 0\text{m})$ and the radius is 20m. In such a setting, the MUs are closer to the AP than to the BS (otherwise, the benefit of traffic offloading is often not significant). We use the similar method in [38] to model the channel power gain, i.e., $g_{iB} = \frac{\varrho_{iB}}{l^{\kappa}_{iB}}$, where l_{iB} denotes the distance between MU i and the BS, and κ denotes the power-scaling factor for the path-loss (we set $\kappa = 3$). We assume that ϱ_{iB} follows an exponential distribution with unit mean due to channel fading. In Appendix V, we further illustrate the performance of our proposed algorithm under more general different topology-settings, including different AP locations and different MUs' location distributions.

System resources: We set the AP's channel bandwidth as $W_A =$ 20MHz. For each MU, we set the BS' minimum bandwidth allocation as $x^{\min}_B = 0.1$MHz and the maximum bandwidth allocation as $x^{\max}_B = 3$MHz (close to a WCDMA channel). For each MU, we set $P^{\max}_{iB} = 0.25$W (i.e., Power Class-3 of mobile devices), $P^{\max}_{iA} = 0.2$W, and $n_0 = 1 \times 10^{-15}$W/Hz [39], [40]. We set the weight $\alpha = 0.02$.

A. Performance of LS-DRhoA-Algorithm to Solve Subproblem (TPA)

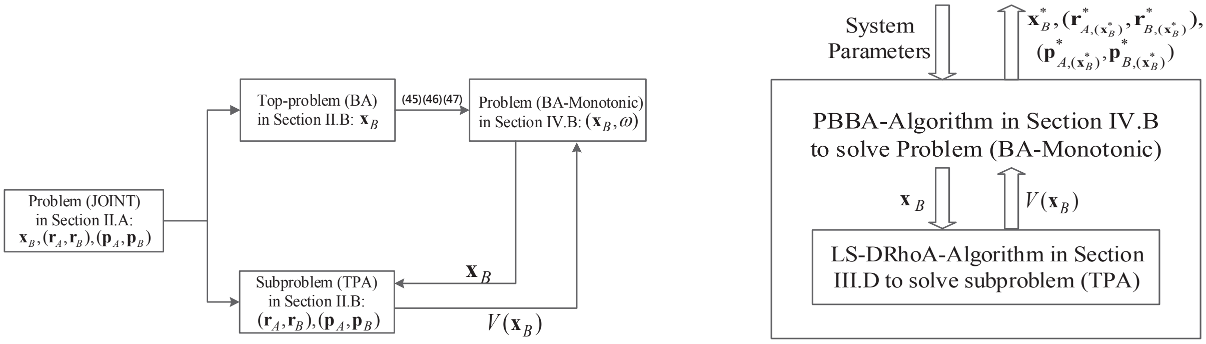
Tables I and II show the accuracy and efficiency of LS-DRhoA-Algorithm in solving subproblem (TPA). We use two scenarios, i.e., the 8-MU scenario (in Table I) and the 16-MU scenario (in Table II), in which the MUs' positions and channel gains are randomly generated as described before. For each scenario, we test $x_{iB} = 1$MHz (in the top subtable) and 2MHz (in the bottom subtable), $\forall i \in \mathcal{I}$. To execute a line-search for the optimal ρ_{0A}, we set the step-size $\Delta = 0.001, 0.0005$ and 0.0001 in LS-DRhoA-Algorithm. As a comparison benchmark, we use the commercial optimization software LINGO [45] to obtain the global optimal solution of subproblem (TPA). Since subproblem (TPA) is a non-convex optimization problem, we need to use LINGO's global-solver to solve it. LINGO's global-solver converts the original non-convex problem into several subproblems, and then uses the branch-and-bound technique to exhaustively search over these subproblems for the global solution (however, the downside of using the LINGO's global-solver is that it might consume a long computational time). Specifically, in each cell of Tables I and II, the first value denotes the optimal objective function value obtained by LS-DRhoA-Algorithm, and the second value denotes the computational time used by LS-DRhoA-Algorithm. For comparison, the last row of the tables show the corresponding results of LINGO's global-solver. All the results are obtained with a PC of Intel(R) Core(TM) i5-4590 CPU@3.3GHz. The results in the three tables show that the optimal value obtained by LS-DRhoA-Algorithm matches well with that obtained by LINGO's global-solver. Even with a moderate step-size $\Delta = 0.001$ (namely, using 1000 samples within $\rho_{0A} \in (0, 1]$), the resulting average difference over all tested cases is no greater than 0.1% (as shown the last column in the tables), which validates the accuracy of our LS-DRhoA-Algorithm. Moreover, Tables I and II also show that our LS-DRhoA-Algorithm consumes a significantly less computational time than LINGO's global-solver (in Tables I and II, for the sake of clear presentation, we use "1hr+" to denote the LINGO's computational time, if LINGO uses more than 1 hour to get the solution). This validates the efficiency of LS-DRhoA-Algorithm. The key reason for this advantage is that our LS-DRhoA-Algorithm exploits the convexity of the problem and adopts an efficient approach, i.e., solving a convex subproblem for each ρ_{0A} along with a line-search of $\rho_{0A} \in (0, 1]$, to find the optimal solution.

The above Tables I and II also show the impact of different Δ on the performance of LS-DRhoA-Algorithm. The tables show that using a smaller step-size Δ helps improve the accuracy of LS-DRhoA-Algorithm, which is consistent with Proposition 5. Nevertheless, the comparison between the three respective rows (for $\Delta = 0.001$, 0.0005, and 0.0001) shows that using $\Delta = 0.001$ already yields the optimal solution of a sufficient accuracy, i.e., less than 0.1% difference compared with LINGO's global-solver. Using an even smaller step-size $\Delta = 0.0001$ yields a very marginal improvement on the accuracy but significantly increases the computational time. Hence, in the rest of this work, we use $\Delta = 0.001$ in LS-DRhoA-Algorithm to execute the remaining numerical examples.

B. Performance of our PBBA-Algorithm to Solve Top-problem (BA)

In Figure 4 and Figure 5, we show the accuracy and efficiency of our PBBA-Algorithm to solve top-problem (BA). For the purpose of comparison, we again use LINGO's global-solver to directly solve Problem (JOINT), which consumes a much longer computational time. Specifically, Figures 4(a), 4(b), 5(a), and 5(b) show the minimum overall system costs obtained by our PBBA-Algorithm compared with those obtained by LINGO's global-solver under different tested cases. Correspondingly, Figures 4(c), 4(d), 5(c), and

[8] In addition to the analysis of the required iteration number of our proposed algorithm, we consider that the overall computational complexity of our algorithm grows quadratically with respect to the number of MUs for the following two reasons. First, each iteration of LS-DRhoA-Algorithm requires I analytical calculations. Second, each round of PBBA-Algorithm generates at most I new vertices, which invoke LS-DRhoA-Algorithm at most I times. Thus, the overall complexity of our proposed algorithm grows quadratically with respect to the number of the MUs I, which is acceptable for a moderately sized group of MUs.


Top-problem (BA)
in Section II.B: $\mathbf{x}_B$
(45)(46)(47)
Problem (BA-Monotonic)
in Section IV.B: $(\mathbf{x}_B, \omega)$
Problem (JOINT)
in Section II.A:
$\mathbf{x}_B, (\mathbf{r}_A, \mathbf{r}_B), (\mathbf{p}_A, \mathbf{p}_B)$
$\mathbf{x}_B$
Subproblem (TPA)
in Section II.B:
$(\mathbf{r}_A, \mathbf{r}_B), (\mathbf{p}_A, \mathbf{p}_B)$
$V(\mathbf{x}_B)$
System
Parameters
$\mathbf{x}^*_B, (\mathbf{r}^*_{A,(\mathbf{x}^*_B)}, \mathbf{r}^*_{B,(\mathbf{x}^*_B)}),$
$(\mathbf{p}^*_{A,(\mathbf{x}^*_B)}, \mathbf{p}^*_{B,(\mathbf{x}^*_B)})$
PBBA-Algorithm in Section IV.B
to solve Problem (BA-Monotonic)
$\mathbf{x}_B$
$V(\mathbf{x}_B)$
LS-DRhoA-Algorithm in Section
III.D to solve subproblem (TPA)

(a) (b)

Fig. 3: Subfigure (a): Connections between top-problem (BA) and Problem (BA-Monotonic) regarding the decomposition of Problem (JOINT). Subfigure (b): Connections between our PBBA-Algorithm to solve Problem (BA-Monotonic) and the subroutine LS-DRhoA-Algorithm to solve subproblem (TPA).

TABLE I: Performance of LS-DRhoA-Algorithm to Solve Subproblem (TPA) (8-MU Scenario)

8MUs: $x_{iB} = 1$MHz	$R_i^{\text{req}} = 4$Mbps	$R_i^{\text{req}} = 4.2$Mbps	$R_i^{\text{req}} = 4.4$Mbps	$R_i^{\text{req}} = 4.6$Mbps	$R_i^{\text{req}} = 4.8$Mbps	$R_i^{\text{req}} = 5$Mbps	Ave. Error
LS-DRhoA ($\Delta = 0.001$)	0.131347, 0.562s	0.204225, 0.515s	0.290057, 0.452s	0.389734, 0.365s	0.504230, 0.249s	0.635106, 0.218s	0.046%
LS-DRhoA ($\Delta = 0.0005$)	0.131347, 1.092s	0.204225, 1.045s	0.290047, 1.014s	0.389696, 0.920s	0.504218, 0.421s	0.635106, 0.452s	0.043%
LS-DRhoA ($\Delta = 0.0001$)	0.131339, 7.535s	0.204217, 7.348s	0.290032, 6.848s	0.389692, 6.677s	0.504201, 3.931s	0.635101, 4.025s	0.039%
LINGO	0.131285, 511s	0.204144, 1371s	0.289916, 1hr+	0.389541, 1hr+	0.504004, 1hr+	0.634839, 1hr+	

8MUs: $x_{iB} = 2$MHz	$R_i^{\text{req}} = 4$Mbps	$R_i^{\text{req}} = 4.2$Mbps	$R_i^{\text{req}} = 4.4$Mbps	$R_i^{\text{req}} = 4.6$Mbps	$R_i^{\text{req}} = 4.8$Mbps	$R_i^{\text{req}} = 5$Mbps	Ave. Error
LS-DRhoA ($\Delta = 0.001$)	0.122560, 0.671s	0.183439, 0.733s	0.250150, 0.624s	0.322908, 0.624s	0.402275, 0.562s	0.488405, 0.515s	0.031%
LS-DRhoA ($\Delta = 0.0005$)	0.122560, 1.279s	0.183438, 1.419s	0.250150, 1.404s	0.322901, 1.186s	0.402275, 1.021s	0.488396, 1.029s	0.030%
LS-DRhoA ($\Delta = 0.0001$)	0.122557, 8.674	0.183430, 9.017s	0.250144, 8.502s	0.322901, 8.096s	0.402265, 7.410s	0.488396, 7.332s	0.028%
LINGO	0.122519, 743s	0.183376, 812s	0.250074, 1hr+	0.322819, 1hr+	0.402156, 1hr+	0.488253, 1hr+	

TABLE II: Performance of LS-DRhoA-Algorithm to Solve Subproblem (TPA) (16-MU Scenario)

16MUs: $x_{iB} = 1$MHz	$R_i^{\text{req}} = 3$Mbps	$R_i^{\text{req}} = 3.2$Mbps	$R_i^{\text{req}} = 3.4$Mbps	$R_i^{\text{req}} = 3.6$Mbps	$R_i^{\text{req}} = 3.8$Mbps	$R_i^{\text{req}} = 4$Mbps	Ave. Error
LS-DRhoA ($\Delta = 0.001$)	1.145647, 0.97s	1.468776, 0.33s	1.849287, 0.33s	2.294140, 0.22s	2.812935, 0.16s	3.426828, 0.14s	0.0036%
LS-DRhoA ($\Delta = 0.0005$)	1.145618, 1.78s	1.468762, 0.67s	1.849275, 0.55s	2.294140, 0.44s	2.812935, 0.31s	3.426828, 0.19s	0.0029%
LS-DRhoA ($\Delta = 0.0001$)	1.145609, 16.19s	1.468713, 9.59s	1.849249, 9.11s	2.294091, 8.23s	2.812888, 5.66s	3.426805, 7.46s	0.0013%
LINGO	1.145574, 1hr+	1.468711, 1hr+	1.849246, 1hr+	2.294038, 1hr+	2.812855, 1hr+	3.426784, 1hr+	

16MUs: $x_{iB} = 2$MHz	$R_i^{\text{req}} = 3$Mbps	$R_i^{\text{req}} = 3.2$Mbps	$R_i^{\text{req}} = 3.4$Mbps	$R_i^{\text{req}} = 3.6$Mbps	$R_i^{\text{req}} = 3.8$Mbps	$R_i^{\text{req}} = 4$Mbps	Ave. Error
LS-DRhoA ($\Delta = 0.001$)	0.813386, 1.79s	0.995905, 1.29s	1.196596, 1.64s	1.416227, 0.45s	1.655739, 0.42s	1.915809, 0.37s	0.0124%
LS-DRhoA ($\Delta = 0.0005$)	0.813364, 2.74s	0.995904, 2.31s	1.196596, 2.07s	1.416227, 2.18s	1.655739, 1.87s	1.915809, 0.75s	0.0119%
LS-DRhoA ($\Delta = 0.0001$)	0.813357, 19.05s	0.995890, 18.51s	1.196587, 17.36s	1.416206, 16.58s	1.655722, 15.65s	1.915791, 13.35s	0.0108%
LINGO	0.813327, 1hr+	0.995642, 1hr+	1.196570, 1hr+	1.416011, 1hr+	1.655658, 1hr+	1.915460, 1hr+	

5(d) show the computational time used by our PBBA-Algorithm compared with those used by LINGO's global-solver.

Figures 4(a), 4(b), 5(a), and 5(b) show that our PBBA-Algorithm achieves the optimal solution with less than 2.5% difference compared with the solution obtained by LINGO's global-solver, thus verifying the accuracy of our PBBA-Algorithm. Figures 4(c), 4(d), 5(c), and 5(d) further verify the efficiency of our PBBA-Algorithm, which saves more than 90% of the computational time used by LINGO's global-solver. Notice that a smaller computational time (comparing with that used by LINGO) means that our proposed algorithm is more favorable for practical applications. For example, when MUs' channel conditions are time-varying due to mobility, a smaller computational time enables our proposed algorithm to converge faster and better track the channel conditions to deliver the maximum performance.

C. Illustration of the Optimal Data Offloading Solution

Figure 6 and Figure 7 demonstrate the optimal data offloading solution. Figure 6(a) illustrates the MUs' optimal traffic scheduling decisions, and Figure 6(b) illustrates the MUs' optimal total power allocations to the AP and BS. Figure 7(a) illustrates the BS' bandwidth usage, and Figure 7(b) illustrates the overall system cost.

Let us first consider Figure 6(a). When the traffic demands are low (e.g., $R_i^{\text{req}} = 5$Mbps), the MUs offload all their traffic demands to the AP and do not transmit any data to the BS. We refer to this case as *full-offloading*. In this case, the MUs take advantage of the relatively short distance to the AP and offload as much traffic as possible to the AP, in order to reduce the power consumption and save the BS' bandwidth usage. When the MUs' traffic demands are large (e.g., $R_i^{\text{req}} = 10$Mbps), however, offloading all traffic to the AP

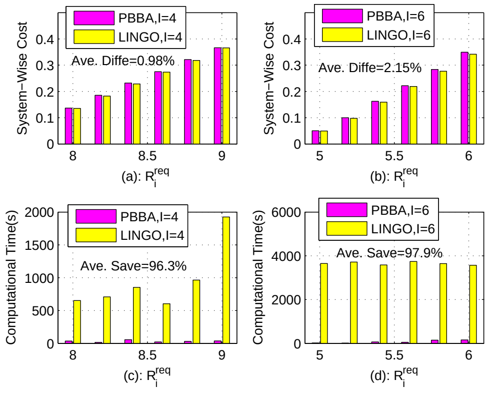


Fig. 4: Performance of PBBA-Algorithm to solve top-problem (BA) for the cases of $I = 4$ and $I = 6$.

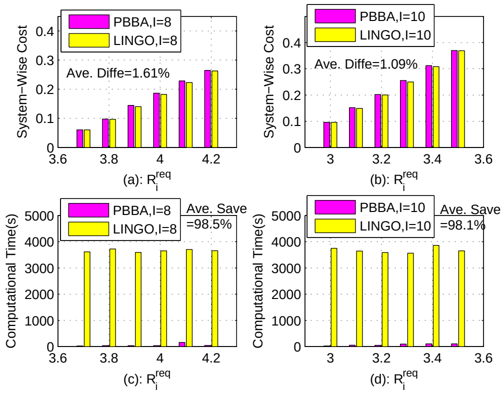


Fig. 5: Performance of PBBA-Algorithm to solve top-problem (BA) for the cases of $I = 8$ and $I = 10$.

will lead to severe co-channel interferences among the MUs, which will significantly increase the MUs' transmit-powers. This explains why the MUs will also deliver part of their traffic to the BS when the demands are large. The numbers above the bar-plots in Figure 6(a) denote the ratios of the offloaded traffic to the total MUs' traffic under different tested cases. The results show that this ratio is non-increasing in the MUs' demands. Figure 6(b) shows a similar trend as Figure 6(a), but in terms of the MUs' transmit-power allocations to the AP and the BS. As the traffic demand increases, the increase of the transmit-power to the BS is much faster than the increase of the transmit-power to the AP. This is due to the long distance between the MUs and BS.

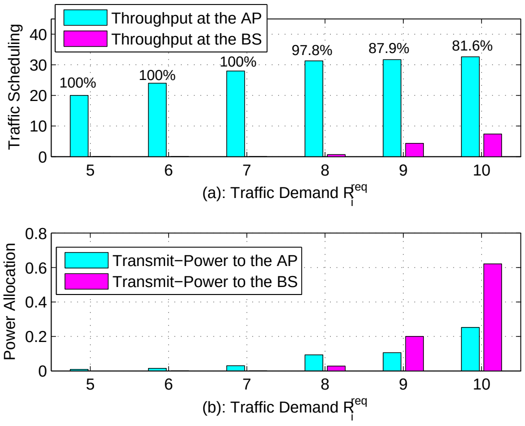


Fig. 6: Optimal offloading solution (MUs' traffic scheduling & power allocation) for the 4-MU scenario. (a): the MUs' throughput to the AP and BS; (b): the MUs' total transmit-power consumptions to the AP and BS.

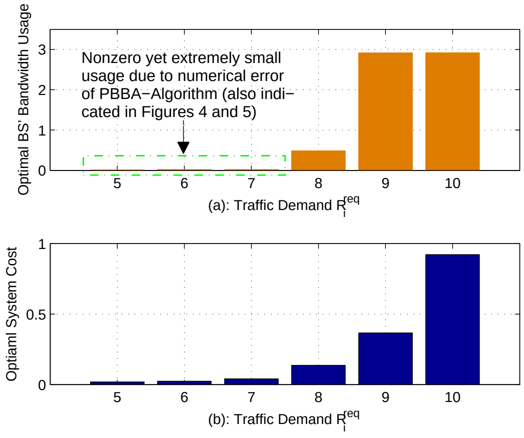


Fig. 7: Optimal offloading solution (the BS' bandwidth usage and overall system cost) for the 4-MU scenario. Subplot (a): the BS' total bandwidth usage for all MUs; Subplot (b): the optimal overall system cost.

Figure 7 further shows the BS' bandwidth usage and the overall system cost. Figure 7(a) shows that the BS consumes almost no bandwidth usage when no MU's data is delivered to the BS (i.e., when the MUs' traffic demands are low)[9]. However, the BS' bandwidth usage increases quickly when the MUs' data delivered to the BS increases (i.e., when the MUs' traffic demands become large). Such a trend is reasonable. When the MUs' traffic demands become large, the MUs need to deliver part of traffic demands to the BS. However, due to the relatively long distance between the MUs and BS, the MUs need to consume significant transmit-powers to deliver traffic to the BS (as shown in Figure 6(b)). Therefore, for saving the overall system cost, the BS increases the bandwidth allocations to the MUs to reduce their power consumptions. Finally, Figure 7(b) shows that the system cost increases when the MUs' traffic demands increase, which is consistent with the intuition.

D. Advantages of Proposed Traffic Offloading Scheme

Our proposed traffic offloading scheme can significantly reduce the overall system cost. We show this advantage in Figures 8 and 9 by comparing with two different baseline schemes .

In Figure 8, we consider a baseline algorithm in which the BS' bandwidth allocation to all MUs is fixed, while the MUs' traffic scheduling and power allocations are optimally computed according to the solution of subproblem (TPA). In other words, the considered baseline algorithm involves a joint optimization of the MUs' traffic scheduling and power allocation, but with the fixed BS' bandwidth allocation. In Figure 8, we have tested the fixed BS' bandwidth allocation as $x_{iB} = x_B^{\max} \times (\frac{1}{8}, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1)$ (which are denoted by $\frac{1}{8}$-BA, $\frac{1}{4}$-BA,$\frac{1}{2}$-BA,$\frac{3}{4}$-BA, and Full-BA, respectively). The comparison results show that our proposed scheme can significantly reduce the overall system cost compared with the baseline algorithm with the fixed BS' bandwidth allocation, which thus validates the importance of optimizing the bandwidth allocation in the offloading process.

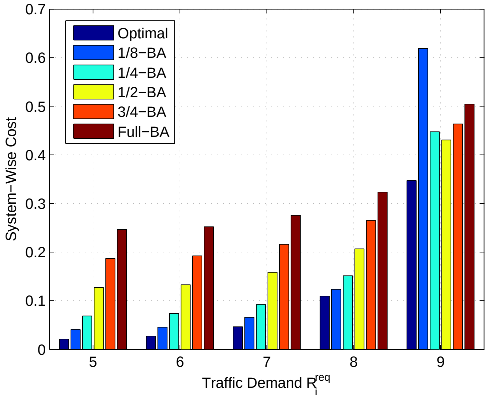


Fig. 8: Performance comparison with the fixed bandwidth scheme.

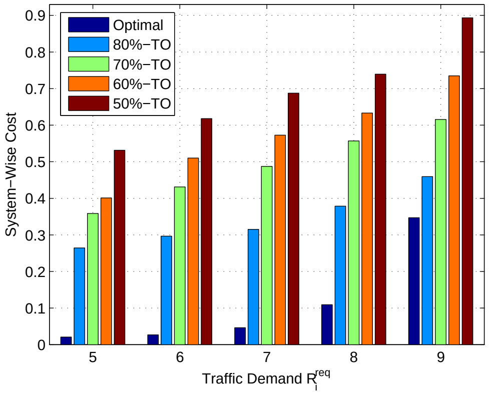


Fig. 9: Performance comparison with the fixed traffic scheduling scheme.

In Figure 9, we consider another baseline algorithm in which the MUs' DC-enabled traffic scheduling between the BS and small cell is fixed, while the MUs' power allocations are optimally computed according to Problem (TPA), and the BS' bandwidth allocation is optimally computed according to Problem (BA). In other words, this baseline algorithm involves a joint optimization of the MUs' power allocations and the BS' bandwidth allocation, but with the fixed MUs' traffic scheduling. Specifically, in Figure 9, we test the MUs' fixed traffic scheduling to small cell as

[9] Since the MU's minimum bandwidth allocation $x_B^{\min}$ (as required in constraint (7)) for signalling and hand-shaking with the BS is usually very small, Figure 7(a) does not include this part of minimum usage for the sake of clear illustration.

$r_{iA} = R_i^{\text{req}} \times (50\%, 60\%, 70\%, 80\%)$ (which are denoted by 50%-TO, 60%-TO,70%-TO, and 80%-TO, respectively). The comparison results show our proposed scheme can significantly reduce the overall system cost compared with the baseline algorithm with the MUs' fixed traffic scheduling, which thus validates the importance of carefully optimizing the MUs' offloaded traffic towards small cell in the DC-enabled offloading.

Figure 10 shows the performance of our proposed traffic offloading scheme under different α, and verifies the tradeoff between the BS' bandwidth usage and the MUs' total power consumption. Figure 10(a) shows that the average BS' bandwidth usage decreases when α increases, since a larger α imposes a larger penalty on the BS' bandwidth usage and thus discourages the BS' bandwidth allocation. As a result, the MUs need to consume larger transmit-powers for offloading data to the AP. Figure 10(b) verifies such a trend and shows that the MUs' total power consumption increases as α increases. Figure 10 demonstrates the intrinsic tradeoff between consuming the BS' spectrum resource and consuming the MUs' power resources. A key motivation of our study here is to take into account the use of these two resources in data offloading and design an optimal scheme to strike a desirable balance between them according to the given weight α. In Appendix VI, we further illustrate the performance of our proposed algorithm under different values of α by comparing with the fixed bandwidth allocation scheme and the fixed traffic scheduling scheme.

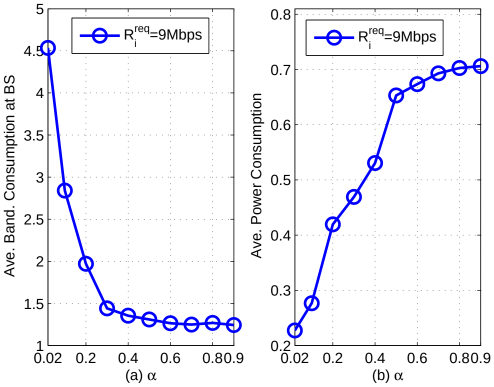


Fig. 10: Performance under different α. We set $I = 4$, and each value in this figure denotes the average result of 100 random sets of the MUs' locations and channel power gains. (a): Average bandwidth usage at the BS; (b): Average total power consumption of all MUs.

VI. Conclusions

In this paper, based on the new paradigm of small-cell DC, we have proposed a novel MUs' uplink traffic offloading scheme through a joint optimization of the BS' bandwidth allocation as well as the MUs' traffic scheduling and power allocation. Our proposed scheme takes into account the MUs' co-channel interferences when offloading data to the AP, and aims at minimizing the overall system cost including both the BS' bandwidth usage and the MUs' total power consumption. Despite the non-convexity of the formulated joint optimization problem, we have designed an efficient algorithm to solve it and compute the optimal offloading solution in a distributed manner. Numerical results show that the proposed algorithm can achieve the global optimum solution with a significantly reduced computational time. Specifically, the proposed algorithm can save more than 90% of the computational time compared with that using LINGO. The numerical results also show that the proposed traffic offloading scheme can significantly reduce the overall system cost, i.e., saving more than 60% of the cost compared with the fixed bandwidth allocation scheme and more than 75% of the cost compared with the fixed traffic scheduling scheme.

In our future work, we will investigate the scenario of many BSs, APs, and MUs. To reduce the implementation-complexity, we need to divide the MUs into multiple subgroups of moderate sizes, and each subgroup will optimize the offloading choices on its own. Our study in this paper provides a useful initial step towards understanding this more general network scenario.

References

[1] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2014-2019," White Paper, Feb. 2015.

[2] A. Aijaz, H. Aghvami, and M. Amani, "A Survey on Mobile Data Offloading: Technical and Business Perspectives," *IEEE Wireless Communications*, vol. 20, no. 2, pp. 104-112, Apr. 2013.

[3] R. Maallaw, *et. al.*, "A Comprehensive Survey on Offload Techniques and Management in Wireless Access and Core Networks," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1582-1604, Third Quarter, 2015.

[4] "Technical specification group radio access network; Study on small cell enhancements for E-UTRA and E-UTRAN; Higher layer aspects (Release 12)," 3rd Generation Partnership Project, Sophia-Antipolis Cedex, France, 3GPP TR 36.842, V12.0.0, Dec. 2013. [Online]. Available: http://www.3gpp.org/ftp/

[5] N.A. Ali, *et. al.*, "Quality of Service in 3GPP R12 LTE-Advanced," *IEEE Communications Magazine*, vol. 51, no. 8, pp. 103-109, Aug. 2013.

[6] S.C. Jha, K. Sivanesan, R. Vannithamby, and A.T. Koc, "Dual Connectivity in LTE Small Cell Networks," in *Proc. of IEEE GLOBECOM'2014*.

[7] A. Mukherjee, "Macro-Small Cell Grouping in Dual Connectivity LTE-B Networks with Non-ideal Backhaul," in *Proc. of IEEE ICC'2014*.

[8] J. Liu. J.G. Liu, and H. Sun, "An Enhanced Power Control Scheme for Dual Connectivity," in *Proc. of VTC-Fall'2014*.

[9] Y. Wu, K. Guo, J. Huang, and X. Shen, "Secrecy-based Energy-Efficient Data Offloading via Dual-Connectivity over Unlicensed Spectrums," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3252-3270, Dec. 2016.

[10] A. Mukherjee, "Optimal Flow Bifurcation in Networks with Dual Base Station Connectivity and Non-ideal Backhaul," in *Proc. of Asilomar Conference on Signals, Systems and Computers*, pp. 521-524, Nov. 2014.

[11] H. Wang, C. Rosa, and K.I. Pedersen, "Dual Connectivity for LTE-advanced Heterogeneous Networks," *Wireless Networks*, Published online Aug. 07, 2015

[12] S. Singh, M. Geraseminko, S.P. Yeh, N. Himayat, and S. Talwar, "Proportional Fair Traffic Splitting and Aggregation in Heterogeneous Wireless Networks", *IEEE Communications Letters*, vol. 20, no. 5, pp. 1010-1013, May 2016.

[13] O. Semiari, W. Saad, and M. Bennis, "Context-Aware Scheduling of Joint Millimeter Wave and Microwave Resources for Dual-Mode Base Stations," in *Proc. of IEEE ICC'2016*.

[14] A. Zakrzewska, D. Lopez-Perez, S. Kucera, and H. Claussen, "Dual Connectivity in LTE HetNets with Split Control- and User-plane", in *Proc. of IEEE GLOBECOM'2013, WORKSHOP*.

[15] J. Huang, *et. al.*, "Joint Scheduling and Resource Allocation in Uplink OFDM Systems for Broadband Wireless Access Networks," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 2, pp. 226-234, Feb. 2009.

[16] Y. Yang, T.Q.S. Quek, L. Duan, "Backhaul-Constrained Small Cell Networks: Refunding and QoS Provisioning," *IEEE Transactions on Wireless Communications*, vol. 13, no. 9, pp. 5148-5161, Sep. 2014.

[17] X. Kang, Y.K. Chia, S. Sun, and H.F. Chong, "Mobile Data Offloading Through A Thrid-Party WiFi Access Point: An Operator's Perspective," *IEEE Transactions on Wireless Communications*, vol. 13, no. 10, pp. 5340-5351, Oct. 2014.

[18] G. Yu, Y. Jiang, L. Xu, and G. Li, "Multi-Objective Energy-Efficient Resource Allocation for Multi-RAT Heterogenous Networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 10, pp. 2118-2127, Oct. 2015.

[19] 3GPP R1-140455, "Physical Layer Aspects for Dual Connectivity", Qualcomm, RAN1#76, Prague, Czech Republic, 2014.

[20] 3GPP R1-140625, "Views on Open Issues for Dual Connectivity", NTT DOCOMO, RAN1#76, Prague, Czech Republic, 2014.

[21] K. Lee, I. Rhee, J. Lee, S. Chong, Y. Yi, "Mobile Data Offloading: How Much Can WiFi Deilver," in *Proc. of ACM CoNEXT'2010*.

[22] S. Dimatteo, P. Hui, B. Han, V.O.K. Li, "Cellular Traffic Offloading through WiFi Networks," in *Proc. of IEEE MASS'2011*.

[23] F. Zhang, W.Y. Zhang, and Q. Ling, "Non-Cooperative Game for Capacity Offload," *IEEE Transactions on Wireless Communications*, vol. 11, no. 4, pp. 1565-1575, Apr. 2012.

[24] Z. Wang and V.W.S. Wong, "A Novel D2D Data Offloading Scheme for LTE Networks," *in Proc. of IEEE ICC'2015*.

[25] Q. Ye, *et. al.*, "User Association for Load Balancing in Heterogeneous Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2706-2716, Jun. 2013.

[26] C.K. Ho, D. Yuan, S. Sun, "Data Offloading in Load Coupled Networks: A Utility Maximization Framework," *IEEE Transactions on Wireless Communications*, vol. 13, no. 4, pp. 1912-1931, Apr. 2014.

[27] Y. Wu, L. Qian, "Energy-Efficient NOMA-enabled Traffic Offloading via Dual-Connectivity in Small-Cell Networks," *IEEE Communications Letters*, *IEEE Communications Letters*, vol.21, no.7, pp.1605-1608, July 2017.

[28] X. Chen, J. Wu, Y. Cai, H. Zhang, T. Chan, "Energy-Efficiency Oriented Traffic Offloading in Wireless Networks: A Brief Survey and A Learning Approach for Heterogeneous Cellular Networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 4, pp. 627-640, Apr. 2015.

[29] G. Iosifidis, L. Gao, J. Huang, L. Tassiulas, "A Double-Auction Mechanism for Mobile Data-Offloading Markets," *IEEE Transactions on Networking*, vol. 23, no. 5, pp. 1634-1647, Oct. 2015.

[30] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong, "Mobile Data Offloading: How Much can WiFi Deliver?" in *IEEE/ACM Transactions on Networking*, vol. 21, no. 2, pp. 536-550, Apr. 2013.

[31] S.C. Liew and Y.J. Zhang, "Proportional Fairness in Multi-Channel Multi-Rate Wireless Networks-Part I: The Case of Deterministic Channels with Application to AP Association Problem in Large-Scale WLAN," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 3446-3456, Sep. 2008.

[32] H. Dahrouj, and W. Yu, "Coordinated Beamforming for the Multi-cell Multi-antenna Wireless Systems", *IEEE Transactions on Wireless Communications*, vol 9, no. 5, pp. 429-434, May 2010.

[33] A. Wiesel, Y.C. Eldar, and S. Shamai, "Linear Precoding via Conic Optimization for Fixed MIMO receivers", *IEEE Transactions on Signal Processing*, vol. 54, no. 1, Jan. 2006.

[34] C. Rosa, K. Pedersen, and H. Wang "Dual connectivity for LTE small cell evolution: Functionality and performance aspects," *IEEE Communications Magazine*, vol. 54, no. 6, pp. 137-143, Jun. 2016.

[35] 3GPP TR36.932, "Scenarios and Requirements for Small Cell Enhancements for E-UTRA and E-UTRAN," v. 12.1.0, Dec. 2013.

[36] Nokia Networks, "Future Work: Optimizing Spectrum Utilisation towards 2020," White Paper, available online at http://networks.nokia.com/file/30301/optimising-spectrum-utilisation-towards-2020.

[37] Executive Summary, "Understanding 3GPP Release 12: Standards for HSPA+ and LTE Enhancements," Feb. 2015, available online at http://www.3gpp.org/specifications/releases/68-release-12

[38] R. Zhang, "Optimal Dynamic Resource Allocation for Multi-Antenna Broadcasting with Heterogeneous Delay-Constrained Traffic," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 2, pp. 243-255, Apr. 2008.

[39] National Instruments, "Introduction to UMTS Device Testing Transmitter and Receiver Measurements for WCDMA Devices," available online at http://download.ni.com/evaluation/rf/Introduction_to_UMTS_Device_Testing.pdf.

[40] S. Trifunovic, A. Picu, T. Hossmann, K.A. Hummel, "Slicing the Battery Pie: Fair and Efficient Energy Usage in Device-to-Device Communication via Role Switching," *in Proc. of ACM CHANT'2013*.

[41] S. Boyd and L. Vandenberghe, "Convex Optimization," Cambridge University Press, 2004.

[42] Y.J. Zhang, L.P Qian, and J. Huang, "Monotonic Optimization in Communication and Networking Systems," *Foundation and Trends in Networking*, Now Publisher, Oct. 2013.

[43] H. Tuy, "Monotonic Optimization: Problems and Solution Approaches," *SIAM Journal of Optimization*, vol. 11, no. 2, pp. 464-494, Nov. 2000.

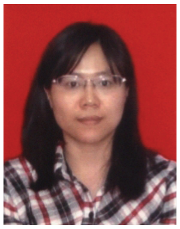
[44] L. Liu, R. Zhang, and K. Chua, "Achieving global optimality for weighted sumrate maximization in the K-user gaussian interference channel with multiple antennas," *IEEE Transactions on Wireless Communications*, vol. 11, no. 5, pp. 1933-1945, May. 2012.

[45] L. Schrage, "Optimization Modeling with LINGO," the 5th edition, Lindo System, Jan. 1999.

[46] E.W. Weisstein, "Bisection," from MathWorld - A Wolfram Web Resource. http://mathworld.wolfram.com/Bisection.html.

Yuan Wu (S'08-M'10-SM'16) received the Ph.D degree in Electronic and Computer Engineering from the Hong Kong University of Science and Technology, Hong Kong, in 2010. He is an Associate Professor in the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. During 2010-2011, he was the Postdoctoral Research Associate at the Hong Kong University of Science and Technology. During 2016-2017, he was with the Broadband Communications Research (BBCR) group, Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests focus on resource management for wireless communications and networks, and smart grid. He is the recipient of the Best Paper Award in IEEE International Conference on Communications (ICC) 2016.

Yanfei He is currently pursuing his M.S. degree in College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. His research interest focuses on resource management for wireless communications and networks, and traffic offloading.



Li Ping Qian (S'08-M'10-SM'16) received the Ph.D. degree in information engineering from The Chinese University of Hong Kong, Hong Kong, in 2010. From 2010 to 2011, she was a Post- Doctoral Research Assistant with the Department of Information Engineering, The Chinese University of Hong Kong. She was a Visiting Student with Princeton University in 2009. She was with Broadband Communications Research Laboratory, University of Waterloo, from 2016 to 2017. She is currently an Associate Professor with the College of Information Engineering, Zhejiang University of Technology, China. Her research interests lie in the areas of wireless communication and networking, cognitive networks, and smart grids. Dr. Qian was a co-recipient of the IEEE Marconi Prize Paper Award in wireless communications in 2011.

Jianwei Huang (F'16) is a Professor and Director of the Network Communications and Economics Lab (ncel.ie.cuhk.edu.hk), in the Department of Information Engineering at the Chinese University of Hong Kong. He received the Ph.D. degree from Northwestern University in 2005, and worked as a Postdoc Research Associate at Princeton University during 2005-2007. He is an IEEE Fellow, a Distinguished Lecturer of IEEE Communications Society, and a Thomson Reuters Highly Cited Researcher in Computer Science. He is the co-author of 9 Best Paper Awards, including IEEE Marconi Prize Paper Award in Wireless Communications in 2011. He has co-authored six books, including the textbook on "Wireless Network Pricing." He received the CUHK Young Researcher Award in 2014 and IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2009. He has served as an Associate Editor of IEEE/ACM Transactions on Networking, IEEE Transactions on Network Science and Engineering, IEEE Transactions on Wireless Communications, IEEE Journal on Selected Areas in Communications - Cognitive Radio Series, and IEEE Transactions on Cognitive Communications and Networking. He has served as the Chair of IEEE ComSoc Cognitive Network Technical Committee and Multimedia Communications Technical Committee. He is the recipient of IEEE ComSoc Multimedia Communications Technical Committee Distinguished Service Award in 2015 and IEEE GLOBECOM Outstanding Service Award in 2010.

Xuemin (Sherman) Shen (M'97-SM'02-F'09) is a Professor and University Research Chair, Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research focuses on resource management, wireless network security, social networks, smart grid, and vehicular ad hoc networks. He is an elected member of IEEE ComSoc Board of Governor, and the Chair of Distinguished Lecturers Selection Committee. Dr. Shen served as the Technical Program Committee Chair/Co-Chair for IEEE Globecom16, Infocom14, IEEE VTC10 Fall, and Globecom07, the Symposia Chair for IEEE ICC10, the Tutorial Chair for IEEE VTC'11 Spring and IEEE ICC08, the General Co-Chair for ACM Mobihoc15, the Chair for IEEE Communications Society Technical Committee on Wireless Communications. He also served/serves as the Editor-in-Chief for IEEE Network, Peer-to-Peer Networking and Application, and IET Communications; a Founding Area Editor for IEEE Transactions on Wireless Communications. Dr. Shen received the Excellent Graduate Supervision Award in 2006, and the Outstanding Performance Award from the University of Waterloo, and the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada. Dr. Shen is a registered Professional Engineer of Ontario, Canada, a Fellow of IEEE, Engineering Institute of Canada, Canadian Academy of Engineering, and Royal Society of Canada, and a Distinguished Lecturer of IEEE Vehicular Technology Society and Communications Society.