Distributed Green Offloading and Power Optimization in Virtualized Small Cell Networks With Mobile Edge Computing

Yulun Cheng, Jun Zhang, *Member, IEEE*, Longxiang Yang, Chenming Zhu, and Hongbo Zhu

***Abstract*—Virtualized small cell networks (SCNs) integrated with mobile edge computing (MEC) is a promising paradigm to provide both wideband access and intensive computation economically for user equipments (UEs) in the scenario of multiple mobile virtual network operators and infrastructure providers. However, the model of offloading in this case is often of high complexity and lacks effective solution. In this paper, by jointly considering offloading, time slice and power allocation, we formulate the energy consumption reduction of the UEs in virtualized SCNs with MEC as a mixed integer nonlinear programming. Our aim is to minimize the total energy consumption of the UEs subject to minimum overall throughput of the network. To solve the problem efficiently, we convert it into a biconvex problem by adding auxiliary variable, which enables the derivation of an efficient iterative algorithm by two subproblems. Towards the first subproblem, we introduce local variables to handle the coupling constraint, and propose an alternating direction method of multipliers (ADMM)-based distributed algorithm, where the closed-form expressions of the optimal solutions in variables updating are derived. For the second subproblem, the closed-form expressions of the optimal solution is also derived. Finally, the effectiveness of the proposed algorithm is demonstrated by extensive simulations with different system configurations.**

***Index Terms*—Mobile edge computing, small cell networks, computation offloading, wireless virtualization, ADMM.**

I. Introduction

WITH the vigorous development of mobile Internet and Internet of Things [1], many innovative wireless data services are emerging, such as automatic driving, virtual reality, and mobile payment [2]. Accordingly, the mobile intelligent terminal gradually replaces the personal computer and becomes the main tool in everyday life. These trends pose great challenges to the conventional mobile cellular networks and cloud computing mode [3]. On one hand, the cell partition and spectrum usage of conventional cellular cannot satisfy the access of large number of mobile intelligent terminals. On the other hand, some services with high real-time requirements are very sensitive to latency, while the traditional cloud computing mode requires that the tasks be transferred to the cloud computing center. Since the center often locates in the core network, the returning of the computation results leads to large latency.

To handle the challenge of large capacity access, the future 5G [4], [5] will mainly adopts small cell networks (SCNs) [6], [7] to absorb large amounts of demands on wireless access. In SCNs, the macro base station (MBS) is mainly employed to cover the whole service area, while multiple small base stations (SBS) are deployed to increase the spatial multiplexing of spectrum, which can improve the spectral efficiency and network capacity. Moreover, because of the closer distance between the SBS and user equipment (UE), the energy consumption caused by distance loss is greatly reduced.

Meanwhile, mobile edge computing (MEC) [8] is a promising technique for enabling massive scale computing from the heavily loaded cellular network, and is being actively standardized by European telecommunications standards institute (ETSI) [9], [10]. Its main idea is to move the cloud computing platform from the core networks to the edge of wireless access network. The proposal of MEC brings the following benefits. Firstly, by deploying computing and storage platforms on the edge of access network, the load pressures of core network and cloud computing center are greatly alleviated. Secondly, MEC greatly shortens the distance of data transmission, which significantly improves the response speed and user experience. Finally, by offloading computing tasks to the MEC servers, UEs greatly reduce their own energy consumption and prolong the battery life.

To some extent, since SBS can be considered as the edge of future 5G networks, it goes without saying that the advantages of the both can be obtained by deploying MEC servers at SBSs. In fact, ETSI has extended MEC further from traditional MBS to the wireless edge devices including SBS, so as to improve network performance [10].

Manuscript received January 15, 2019; revised July 25, 2019; accepted October 17, 2019. Date of publication October 24, 2019; date of current version March 18, 2020. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC1314903; in part by the Natural Science Foundation of China under Grant 61601238, Grant U1805262, Grant 61671251, and Grant 61871446; and in part by the China Postdoctoral Science Foundation under Grant 2019M661773. The associate editor coordinating the review of this article and approving it for publication was X. Zhang. *(Corresponding author: Yulun Cheng.)*

Y. Cheng is with Institute of Communications and Information Technology, China Information Consulting and Designing Institute Company Ltd., Nanjing 210003, China, and also with the Jiangsu Key Laboratory of Wireless Communications, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (e-mail: chengyuluen@163.com).

J. Zhang, L. Yang, and H. Zhu are with the Jiangsu Key Laboratory of Wireless Communications, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (e-mail: zhangjun@njupt.edu.cn; yanglx@njupt.edu.cn; zhuhb@njupt.edu.cn).

C. Zhu is with China Information Consulting and Designing Institute Company Ltd., Nanjing 210003, China (e-mail: zhuchenming@cicdi.com).

Digital Object Identifier 10.1109/TGCN.2019.2949339

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

However, deploying MEC servers on SBSs will inevitably lead to an increase in construction and operating expenses of the infrastructure providers (InPs). In recent years, following its success in reducing the operation and capital expenses, wireless virtualization [11] has been considered as the distinct feature of future 5G networks. The efficacy of wireless virtualization relies on decoupling the physical wireless network infrastructure and spectrum resources from the InPs, and abstracting them into various customized network slices. The slices are shared by multiple mobile virtualized network operators (MVNOs), so the innovative services can be deployed rapidly and flexibly with less costs [12], [13]. By introducing wireless virtualization to MEC aided SCNs, SBSs and MEC servers can be virtualized and shared among multiple MVNOs, so that the service deployment and operating costs of the InPs are greatly reduced.

To achieve a desirable tradeoff between user experience, energy consumption and operating costs in virtualized SCNs with MEC, the offloading problem should be well handled, which has critical impact on the above metrics. The concept of offloading [14] is utilized to address the resource allocation, task and UE association in mobile computing. By far, it is often formulated as a resource allocation problem and solved centrally by optimization theory. Although dedicated works have been done on this area, these results are not very suitable for the virtualized SCNs with MEC, because in this scenario, the model should includes multiple MVNOs, UEs and InPs, which indicates that the centralized solutions are inefficient because of high complexity and expense of overhead.

In this paper, we consider a virtualized SCN scenario, where each SBS is integrated with one MEC server, while these resources are virtualized into multiple slices to allocate among multiple UEs, MVNOs and InPs. We concentrate exclusively on the joint UE association, time slice and power optimization in a distributed manner, so as to minimize the energy consumption of the UEs effectively. The main contributions of the paper are summarized as follows.

- The minimization of energy consumption for all the UEs is formulated as a mixed integer nonlinear programming (MINLP), by taking both the task transmission and local computation into consideration. The formulation is flexible to handle offloading, time slice and power optimization among multiple UEs, MVNOs and InPs, subject to access capacity and minimum overall throughput constraints.
- To handle the formulated problem efficiently, it is converted into a biconvex problem by adding auxiliary variable. And then, an efficient iterative algorithm is proposed by decomposing the biconvex problem into two subproblems. For the first subproblem, local variables are introduced to handle the coupling constraint and convert the problem into the desired form, by which an alternating direction method of multipliers (ADMM)-based distributed algorithm is proposed.
- The closed-form expressions of the optimal solutions for the variables updating in each subproblems are derived, which are utilized to reduce the complexity of the proposed algorithm.
- Extensive simulations are conduced to evaluate the performance, convergence, and the complexity of the proposed algorithm by comparing with the other baseline algorithms.

The reminder of this paper is organized as follows. Section II concludes the related works. Section III presents the system model and problem formulation. To solve the formulated problem, we propose a distributed algorithm in Section IV. Simulation results and discussions are demonstrated in Section V. Finally, Section VI concludes this study.

II. Related Works

While reducing latency is a critical concern of offloading [16], [17], the high complexity makes the problem more intricate in multi-user scenario. In [18], the computation offloading in multi-channel wireless interference is formulated as the multi-user offloading game, and a distributed algorithm is proposed to achieve Nash equilibrium. To improve the transmission rate, the offloading in multiple-input multiple-output networks is studied in [19], in which the UE and the BS are equipped with multiple antennas. The radio and computation resources are jointed assigned according to the solution of the formulated optimization model. To reduce the complexity, the authors proposed a successive convex approximation, which can converge to the local optimal solution of the original problem. For the dynamic energy supply case, the offloading problem is addressed for the energy harvested UE in [20]. The proposed algorithm requires no statical information, so the overhead expense is greatly reduced. In [21], the offloading for the multiple-user is studied with time-division multiple access (TDMA) and orthogonal frequency division multiple access (OFDMA), respectively. To solve the mixed integer optimization for the OFDMA case, a sub optimal algorithm is proposed by transferring the OFDMA problem to the TDMA case, so that the energy consumption can be reduced based on the closed-form threshold. However, all these previous works consider the multi-user and single MEC server assumption, and do not involve the multi-user multi-MEC server scenario.

In fact, for the virtualized SCNs with MEC, the offloading problem has inherent high complexity due to the consideration of multiple users, multiple MEC servers and multiple InPs. Some existing works have made good attempt in this direction, for example, in [22] the authors investigated the task offloading problem in SCNs in the context of wireless virtualization and software defined network, in which the multi-MEC server is considered. To solve the task offloading problem efficiently, a scheme is proposed by dividing the formulated problem into two sub problems. However, its solution is based on the centralized optimization, while our method is on the basis of alternating direction method of multipliers (ADMM) algorithm in a distributed manner. Moreover, in [22], the objective is to minimize the latency, while in our study, the energy consumption is the optimized metric. In the context of ultra dense Internet of Things networks, the offloading problem for multi-MEC server is also investigated in [23]. To avoid the high complexity, game theory and greedy methods are utilized to solve the problem, and a two-tier game theoretic greedy

TABLE I
COMPARISON WITH EXISTING WORKS

Work	[22]	[23]	[28]	[24, 25]	[26, 27]	Proposed
Optimization objective	delay	delay && energy	delay && energy	user rate && operating cost	energy	energy
Centralized or Distributed	C	C	D	D	D	D
Heuristic or deterministic	deterministic	heuristic, greedy	deterministic, BP	deterministic	heuristic, GA, PSO	deterministic, ADMM
Multiple MEC	Y	Y	Y	Y	Y	Y

offloading algorithm is developed. The scheme still belongs to the processing in the centralized manner, and the solution is based on the heuristics, where as in our study, the proposed algorithm is in the distributed manner and on the basis of deterministic optimization theory. Tan *et al.* [24], [25] proposed an offloading strategy for the scenario of multiple users, multiple MEC servers, and multiple MVNOs in SCNs. To reduce the complexity, ADMM algorithm is adopted to solve the formulated mixed integer programming in the distributed manner. Differently, the optimization objective in [24], [25] is to maximize the difference between user rate and operating costs, where as our study mainly focus on the minimizing energy consumption of the UEs. In [26], [27], Guo *et al.* also investigated the energy consumption of the users in SCNs with MEC, and formulated the offloading decision, computation chip, and wireless channel allocation jointly as a MINLP. The proposed algorithm in [26], [27] solve the problem centrally, and is based on the heuristic methods, such as genetic algorithm and particle swarm optimization, where as our method provide a distributed way to obtain the optimal solution, which is mainly based on the closed form expression derived by Karush-Kuhn-Tucher (K. K. T.) contribution. For the distributed offloading in SCNs, Li *et al.* also investigated the energy consumption in [28]. They formulated the optimization objective as the weighted sum of energy and latency, and proposed a belief propagation algorithm to obtain the task allocation. Different from that, in our study, the distributed algorithm is on the basis of ADMM algorithm. Moreover, in [28] partial user associations are predetermined by the positional relationship, where as in our study, it is totally determined by the output of the proposed algorithm.

Table I summarizes the difference of the proposed algorithm from the existing works.

III. System Model and Problem Formulation

A. System Model

As depicted in Fig. 1, we consider the uplink of virtualized SCNs with MEC, which includes InP part, MVNO part, and wireless virtualization part. InP part consists of *M* InPs, and let $M_{\text{InP}} = \{1, \ldots, M\}$ denote the set of all InPs. Each InP $i \in M_{\text{InP}}$ owns one MBS and a set of SBSs S_i. Each SBS $j \in S_i$ is integrated with one MEC server. MVNO part includes *N* MVNOs, and let $N_{\text{MVNO}} = \{1, \ldots, N\}$ denote the set of all MVNOs. Each MVNO $n \in N_{\text{MVNO}}$ has a set of subscribed UEs U_n. Each UE $u \in U_n$ locates randomly in the area covered by the InPs, and subscribes its services from MVNO *n*. We assume that all the SBSs and UEs are equipped with single antenna. Wireless virtualization part consists of the slicing of the SBSs and MEC servers owned by the InPs and the allocation of the slices among the MVNOs. Let *t* denote the uplink duration of each SBS, and we assume that all the SBSs are synchronized [25], [32]. At the beginning of each uplink duration, each UE has one task to be computed, which is split into two parts. One part is computed by UE itself, while the other part is transmitted through one SBS and computed by the corresponding MEC server. It is assumed that each UE *u* can communicate to all the SBSs of the InPs, but it only offloads its task to a particular SBS and MEC server, which is selected by its MVNO. Each UE employs the time slices of the SBSs and the computation slices of MEC servers by its MVNO, while each MVNO pays the rents to the InPs for the utilization of their network resources.

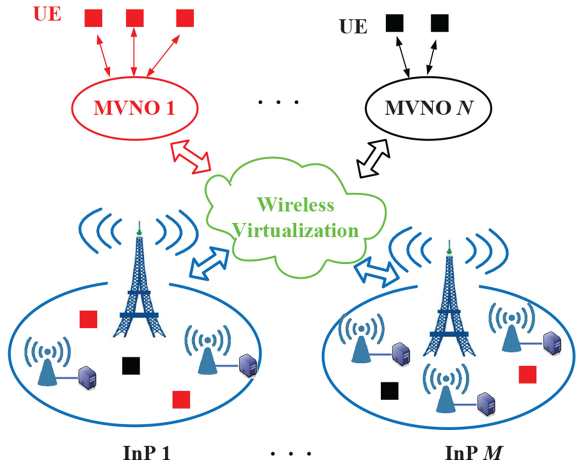

Fig. 1. System model of virtualized small cell networks with MEC.

B. Problem Formulation

The total energy consumption of each UE *u* is formulated as the sum of E_u^{T} and E_u^{L}. E_u^{T} is the energy consumption from task transformation of UE *u*. Denote $a_{u,j}$ as the transmission time of UE *u* to SBS *j*, and let $x_{u,j}$ be the association for UE *u* and SBS *j*, i.e., $x_{u,j} = 1$ when UE *u* is associated with SBS *j*, otherwise, $x_{u,j} = 0$. Thus, E_u^{T} can be expressed as

$$E_u^{\text{T}} = \sum_{i=1}^{M} \sum_{j \in S_i} P_u a_{u,j} x_{u,j}, \tag{1}$$

where P_u denotes the transmission power of UE *u*. E_u^{L} represents the energy consumption from local computing of UE *u*, which can be written as

$$E_u^{\text{L}} = \left(R_u - \sum_{i=1}^{M} \sum_{j \in S_i} r_{u,j} a_{u,j} x_{u,j} \right) P_c, \tag{2}$$

where P_c is the energy consumption per bit by local computing, and R_u is the initial data amount of its task. $r_{u,j}$ denotes the channel rate from UE *u* to SBS *j*, and can be expressed as

$$r_{u,j} = B \log_2 \left(1 + \frac{P_u h_{u,j}}{N_0 + I} \right), \tag{3}$$

where *B* is the spectrum bandwidth, and $h_{u,j}$ is the channel gain from UE *u* to SBS *j*. Here, $h_{u,j} = h d_{u,j}^{-\varphi}$, where *h* is the norm of zero-mean, independent, circular-symmetric complex Gaussian random variable with variances 1, $d_{u,j}$ is the distance from UE *u* to SBS *j*, and φ is the path loss factor. N_0 is the power of the background noise, and *I* denotes the maximum tolerable interference level [33], [34]. Thus, the considered task offloading problem can be formulated as

$$\text{(P1)} \quad \min_{a_{u,j},\ x_{u,j},\ P_u} \quad f_{\text{P1}} = \sum_{n=1}^{N}\sum_{u\in U_n}\left(E_u^{\text{L}} + E_u^{\text{T}}\right) \tag{4}$$

$$\text{s.t.} \quad 0 \le P_u \le P_{\max},\ \forall u \in U_n, \forall n \in N_{\text{MVNO}}, \tag{5}$$

$$\sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i} a_{u,j} r_{u,j} \ge R_{\text{T}}, \tag{6}$$

$$\sum_{i=1}^{M}\sum_{j\in S_i} x_{u,j} = 1,\ \forall u \in U_n, \forall n \in N_{\text{MVNO}}, \tag{7}$$

$$\sum_{n=1}^{N}\sum_{u\in U_n} a_{u,j} \le t,\ \forall j \in S_i, \forall i \in M_{\text{InP}}, \tag{8}$$

$$\sum_{i=1}^{M}\sum_{j\in S_i} a_{u,j} r_{u,j} \le R_u,\ \forall u \in U_n, \forall n \in N_{\text{MVNO}}, \tag{9}$$

$$a_{u,j} \le x_{u,j} t,\ \forall u, j, \tag{10}$$

$$x_{u,j} \in \{0, 1\},\ \forall u, j, \tag{11}$$

$$a_{u,j} \ge 0,\ \forall u, j. \tag{12}$$

Objective function (4) is to minimize the overall energy consumption for all the UEs by jointly optimizing the transmission slot variable $a_{u,j}$, association variable $x_{u,j}$, and power variable P_u. Constraint (5) states the transmission power of each UE cannot exceed the maximum threshold $P_{\max}$ and is no less than 0. Constraint (6) guarantees that the minimum overall throughput of the wireless network is no less than the rate threshold R_{T}. Constraint (7) indicates that each UE offload its task to only one MEC servers. Constraint (8) states that for each MEC server, its total reception duration should not exceed the uplink duration of the InPs. Constraint (9) indicates that for each UE, its offloaded data amount should not exceed the original task amount. Constraint (10) states that only when a MEC server is associated with a UE, can the corresponding transmission duration be non-zero, otherwise, it equals 0. Constraint (11) claims the range of the binary variables.

IV. Problem Solving via Distributed Optimization

Since objective function (4) contains the product of binary variable $x_{u,j}$ and log function of P_u, it makes problem P1 combinatorial. Besides, due to constraint (6), each $a_{u,j}$ and P_u are coupling together, which makes the problem indecomposable. These properties make the optimization of P1 computationally infeasible for large network and hard to provide useful insight. Inspired by the efficient methods of handling joint optimization to MINLP [25], [27], [29], [32], we introduce auxiliary variable to convert the original problem into a biconvex optimization [35] when discarding integer variable $x_{u,j}$ and the involved constraints. And then, an alternate convex search [35]-based framework is proposed to exploit the convex substructure and solve the biconvex problem efficiently. In the proposed framework, the biconvex problem is decomposed into two subproblems with integer variable and constraints previously discarded. By iterating the optimal solutions of the two subproblems mutually, the optimized variables will converge, which is considered as the approximate solution of P1.

A. Biconvexity Transformation of Problem P1

When discarding $x_{u,j}$ and the involved constraints, P1 can be converted to a biconvex problem equivalently. To show this, we introduce auxiliary variable $y_{u,j}$, which denotes the slice of the uplink duration allocated to UE *u* by SBS *j*. Note that $a_{u,j}$ and $y_{u,j}$ are the transmission time from UE *u* and SBS *j*, respectively. If $y_{u,j} < a_{u,j}$, partial transmission energy of UE *u* will be wasted since SBS *j* only allocates $y_{u,j}$ to UE *u*. If $y_{u,j} > a_{u,j}$, partial slice will be idle and does not produce energy consumption, because UE *u* only transmits by length of $a_{u,j}$. Hence, these two cases are excluded while our formulation focuses on case $y_{u,j} = a_{u,j}$. Thus, the energy consumptions of all the UEs from task transmission and local computation can be expressed respectively as

$$f_1\left(y_{u,j}, P_u\right) = \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i} y_{u,j} P_u, \tag{13}$$

$$f_2\left(a_{u,j}, P_u\right) = \sum_{n=1}^{N}\sum_{u\in U_n}\left(R_u - \sum_{i=1}^{M}\sum_{j\in S_i} r_{u,j} a_{u,j}\right) P_c. \tag{14}$$

Thus, by discarding $x_{u,j}$ and the involved constraints, P1 is rewritten as

$$\text{(P2)} \quad \min_{P_u,\ y_{u,j},\ a_{u,j}} \quad f_1\left(y_{u,j}, P_u\right) + f_2\left(a_{u,j}, P_u\right) \tag{15}$$

$$\text{s.t.} \quad y_{u,j} = a_{u,j},\ \forall u, j, \tag{16}$$

$$y_{u,j} \in \Phi_1,\quad \forall u, j, \tag{17}$$

$$a_{u,j} \in \Phi_2,\quad \forall u, j, \tag{18}$$

$$P_u \in \Phi_3,\quad \forall u.$$

where $\Phi_3 = \{P_u |\ 0 \le P_u \le P_{\max}\}$,

$$\Phi_1 = \left\{ y_{u,j} |\ y_{u,j} \ge 0 \ \ \&\&\ \sum_{k=1}^{N}\sum_{u\in U_k} y_{u,j} \le t \right\}, \tag{19}$$

$$\Phi_2 = \left\{ a_{u,j} |\ a_{u,j} \ge 0 \ \ \&\&\ \sum_{i=1}^{M}\sum_{j\in S_i} a_{u,j} r_{u,j} \le R_u \right. \\ \left. \&\&\ \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i} a_{u,j} r_{u,j} \ge R_{\text{T}} \right\}. \tag{20}$$

Note that with constraint (16), it is easy to derived that the objective function of P2 equals that of P1.

Lemma 1: Given $P_u = \tilde{P_u} \in \Phi_3$, problem P2 is convex in $y_{u,j}$ and $a_{u,j}$, respectively.

Proof: See Appendix A. ■

Lemma 2: Given $y_{u,j} = \tilde{y}_{u,j} \in \Phi_1$, problem P2 is convex in P_u.

Proof: See Appendix B. ■

Lemma 3: Given $a_{u,j} = \tilde{a}_{u,j} \in \Phi_2$, problem P2 is convex in P_u.

Proof: See Appendix C. ■

Proposition 1: Problem P2 is biconvex for $P_u \in \Phi_3$ and $y_{y,j} \in \Phi_1$, meanwhile, it is also biconvex for $P_u \in \Phi_3$ and $a_{y,j} \in \Phi_2$.

Proof: According to the definition of biconvex [35], a function $f(x, y)$:$\boldsymbol{X} \times \boldsymbol{Y} \rightarrow \mathbb{R}$ is biconvex if and only if $f(x, y)$ is convex in *y* given $x \in \boldsymbol{X}$ and convex in *x* given $y \in \boldsymbol{Y}$. Combining Lemmas 1 and 2, it can be derived by the definition that P2 is biconvex for $P_u \in \Phi_3$ and $y_{y,j} \in \Phi_1$. Similarly, by combining Lemmas 1 and 3, it is also proved that P2 is biconvex for $P_u \in \Phi_3$ and $a_{y,j} \in \Phi_2$. Therefore, the Proposition is proved. ■

Since P2 is biconvex, a couple of methods [35] can be utilized to solve it efficiently. Here, we develop an alternate convex search [35]-based framework, which exploit the convex substructure of the problem. The framework decomposes P2 into two subproblems by dividing the variables into P_u and $(a_{u,j}, y_{u,j})$. The first sub problem is to optimize $(a_{u,j}, y_{u,j})$ under fixed transmission power P_u via ADMM, where $x_{u,j}$ and the involved constraints are added in the iteration. The second one is to optimize P_u under obtained $(a^*_{u,j}, y^*_{u,j})$. The details are presented in the following subsections.

B. Solving $(a_{u,j}, y_{u,j})$ Under Fixed Transmission Power P_u via ADMM

For any fixed P_u, P2 degenerates into the one of jointly optimizing $(a_{u,j}, y_{u,j})$ in $f_1(y_{u,j}) + f_2(a_{u,j})$ over constraint (16) and sets Φ_1, Φ_2. With Lemmas 1 and 2, it can be obtained that Φ_1 and Φ_2 are both convex sets while $f_1(y_{u,j}) + f_2(a_{u,j})$ and (16) are also convex. Hence, the degenerated problem can be solved efficiently via ADMM [15], which is well suited to distributed convex optimization. Note that objective function $f_1(y_{u,j}) + f_2(a_{u,j})$ and constraint (16) are compatible with the form of [15, eq. (3.1)] by taking $\mathbf{A} = \mathbf{I}$, $\mathbf{B} = -\mathbf{I}$ and $\mathbf{C} = 0$. Therefore, to obtain the desired form, we retain $f_1(y_{u,j}) + f_2(a_{u,j})$ and constraint (16) while bring the remaining constraints into the follow-up, according to which the augmented Lagrange function can be expressed as

$$\begin{aligned} &L_\rho\big(y_{u,j},\ a_{u,j},\ \lambda_{u,j}\big) \\ &= \sum_{n=1}^{N}\sum_{u\in U_n}\left(R_u - \sum_{i=1}^{M}\sum_{j\in S_i} r_{u,j}\, a_{u,j}\right)P_c \\ &\quad + \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i} P_u y_{u,j} \\ &\quad + \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i} \lambda_{u,j}\big(a_{u,j} - y_{u,j}\big) \\ &\quad + \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i} \frac{\rho}{2}\big(a_{u,j} - y_{u,j}\big)^2, \end{aligned} \tag{21}$$

where $\lambda_{u,j}$ are the Lagrange multipliers associated with constraint (16) and ρ is penalty factor.

With (21) and ADMM procedure [15], the distributed optimization iteration can be expressed as follows. Starting by initializing $(y^{(0)}_{u,j}, a^{(0)}_{u,j}, \lambda^{(0)}_{u,j}), k = 0$, the solution of the *k*-th iteration $(y^{(k)}_{u,j}, a^{(k)}_{u,j}, \lambda^{(k)}_{u,j})$ is known, then the following three updating processes are utilized to obtain $(y^{(k+1)}_{u,j}, a^{(k+1)}_{u,j}, \lambda^{(k+1)}_{u,j})$.

a) *Updating $a^{(k+1)}_{u,j}$:* By solving augmented Lagrange function (21), $a^{(k+1)}_{u,j}$ is obtained with given $(y^{(k)}_{u,j}, \lambda^{(k)}_{u,j})$ as

$$\min_{a_{u,j}} \quad L_\rho\Big(a_{u,j},\ y^{(k)}_{u,j},\ \lambda^{(k)}_{u,j}\Big) \tag{22}$$

$$\text{s.t.} \quad a_{u,j} \in \Phi_2,\ \ \forall u, j. \tag{23}$$

where $L_\rho(a_{u,j},\ y^{(k)}_{u,j},\ \lambda^{(k)}_{u,j}) = \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i} [\frac{\rho}{2}(a_{u,j} - y^{(k)}_{u,j})^2 - r_{u,j}\, a_{u,j} P_u + \lambda^{(k)}_{u,j}(a_{u,j} - y^{(k)}_{u,j})]$. However, due to the coupling constraint (6) in Φ_2, the above problem is indecomposable for each MVNO $n \in N_{\mathrm{MVNO}}$. To handle this issue, we introduce the local copy $\boldsymbol{a_n}$ of the related global variable $\boldsymbol{a}$ for each MVNO *n*. Note that $\boldsymbol{a_n} = [\ldots, a_{n,u,j}, \ldots, a_{n,|\sum_{n\in N_{\mathrm{MVNO}}} U_n|,|\sum_{i\in M_{\mathrm{InP}}} S_i|}]$ and $\boldsymbol{a} = [\ldots, a_{u,j}, \ldots, a_{|\sum_{n\in N_{\mathrm{MVNO}}} U_n|,|\sum_{i\in M_{\mathrm{InP}}} S_i|}]$. $\boldsymbol{a_n}$ can be interpreted as MVNO *n*'s opinion about the global variable $\boldsymbol{a}$. Besides, we introduce indicator function $d(\boldsymbol{a_n})$ such that $d(\boldsymbol{a_n}) = 0$ when $a_{n,u,j} \in \Phi_2$, otherwise, $d(\boldsymbol{a_n}) = +\infty$. With these definitions, (22)-(23) are equivalent to

$$\min_{\boldsymbol{a_n}} \quad L_\rho\Big(a_{n,u,j},\ y^{(k)}_{u,j},\ \lambda^{(k)}_{u,j}\Big) + d(\boldsymbol{a_n}) \tag{24}$$

$$\text{s.t.} \quad a_{n,u,j} = a_{u,j},\ \ \forall u, j. \tag{25}$$

Note that objective function (24) and the feasible set are separable for each MVNO, as well as the integer constraints (7), (10), (11). Therefore, problem (24) is decomposed into multiple subproblems solved by each MVNO individually to reduce complexity. That is, for $\forall u \in U_n$, MVNO *n* solves the following problem,

$$\min_{a_{n,u,j},\ x_{u,j}} \quad L'_\rho\Big(a_{n,u,j},\ y^{(k)}_{u,j},\ \lambda^{(k)}_{u,j}\Big) \tag{26}$$

$$\text{s.t.} \quad a_{n,u,j} = a_{u,j},\ \ \forall u, j, \tag{27}$$

$$a_{n,u,j} \in \Phi_2,\ \ \forall u, j, \tag{28}$$

$$(7), (10), (11).$$

where $L'_\rho(a_{n,u,j},\ y^{(k)}_{u,j},\ \lambda^{(k)}_{u,j}) = \sum_{i=1}^{M}\sum_{j\in S_i} [\frac{\rho}{2} a^2_{n,u,j} + (\lambda^{(k)}_{u,j} - r_{u,j} P_c - \rho y^{(k)}_{u,j}) a_{n,u,j} + d(\boldsymbol{a_n})]$. By taking (27) as the linear constraint of [15, eq. (3.1)], the above problem can be rewritten as

$$\begin{aligned} \min_{a_{n,u,j},\ x_{u,j}} \ & L'_\rho\Big(a_{n,u,j},\ y^{(k)}_{u,j},\ \lambda^{(k)}_{u,j}\Big) \\ & + \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i}\Big[\lambda_{n,u,j}\big(a_{n,u,j} - a_{u,j}\big) \\ & \qquad + \frac{\rho_s}{2}\big(a_{n,u,j} - a_{u,j}\big)^2\Big] \\ \text{s.t.} \ & (7),\ (10),\ (11),\ (28). \end{aligned}$$

where $\lambda_{n,u,j}$ are the Lagrange multipliers associated with constraint (27), ρ_s is the corresponding penalty factor. It is easy to derive that the above problem is convex, so we launch an inner ADMM iteration to solve it as

$$a_{n,u,j}^{k'+1} = \arg\min_{a_{n,u,j}} L'_\rho\Big(a_{n,u,j},\ y_{u,j}^{(k)},\ \lambda_{u,j}^{(k)}\Big) + \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i}\Big[\lambda_{n,u,j}^{k'}\Big(a_{n,u,j} - a_{u,j}^{k'}\Big) + \frac{\rho_s}{2}\Big(a_{n,u,j} - a_{u,j}^{k'}\Big)^2\Big], \quad (29)$$

$$a_{u,j}^{k'+1} = \arg\min_{a_{u,j}} \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i}\Big[\lambda_{n,u,j}^{k'}\Big(a_{n,u,j}^{k'+1} - a_{u,j}\Big) + \frac{\rho_s}{2}\Big(a_{n,u,j}^{k'+1} - a_{u,j}\Big)^2\Big], \quad (30)$$

$$\lambda_{n,u,j}^{k'+1} = \lambda_{n,u,j}^{k'} + \rho_s \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i}\Big(a_{n,u,j}^{k'+1} - a_{u,j}^{k'+1}\Big). \quad (31)$$

where k' is the inner iteration index. Given an initial value $(a_{n,u,j}^0, a_{u,j}^0, \lambda_{n,u,j}^0)$ and stop criterion, the inner iteration is employed to obtained $a_{u,j}^*$.

It can be observed that (29) is untraceable, since problem (26) still belongs to MINLP. Thus, to solve it efficiently, we propose a proposition based on the traversal search to obtain the closed-form expression of the optimal solution. The idea behind the proposition is as follows. Firstly, $x_{u,j}$ is fixed by utilizing the feature of constraint (7). Substituting fixed $x_{u,j}$ into problem (26), it can be solved efficiently by the proposed proposition. After that, it moves to the next value of $x_{u,j}$ through traversal search. Note that due to the proposed decomposition, the number of $x_{u,j}$ has been reduced from $|\sum_{u\in U_n}\sum_{n\in N_{\text{MVNO}}} U_n||\sum_{i\in M_{\text{InP}}} S_i|$ to $|U_n||\sum_{i\in M_{\text{InP}}} S_i|$.

Proposition 2: For each $u \in U_n, j \in S_i, i \in M_{\text{InP}}$, the optimal solution $a_{u,j}^*$ for problem (26) can be obtained as

$$a_{u,j}^* = \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i} a_{n,u,j}^* / |\sum_{u\in U_n}\sum_{n\in N_{\text{MVNO}}} U_n||\sum_{i\in M_{\text{InP}}} S_i|. \quad (32)$$

where

$$a_{n,u,j}^* = \begin{cases} a_{n,u,z}^*, & j = \arg\min_{z\in S_i, i\in M_{\text{InP}}} L'_\rho(a_{n,u,z}) \\ 0, & \text{otherwise.} \end{cases} \quad (33)$$

$a_{n,u,z}^*$ is expressed at the top of next page, where $b = (\lambda_{u,j}^{(k)} - \rho_s a_{u,j}^{k'} - r_{u,z}P_c - \rho y_{u,j}^{(k)} + \sum_{n=1}^{N}\sum_{u\in U_n}\sum_{i=1}^{M}\sum_{j\in S_i}\lambda_{n,u,j}^{k'})/\rho$.

Proof: See Appendix D. ■

Algorithm 1 For Updating $a_{u,j}^{(k+1)}$

```
Input: $\lambda_{u,j}^{(k)}, a_{u,j}^{(k)}, y_{u,j}^{(k)}, r_{u,j}, P_c, R_u, t, \rho_s, \varepsilon_0, L = \emptyset, u \in U_n, j \in S_i, i \in M_{\text{InP}}$.
Iteration $k + 1$:
for $z = 1$:$|\sum_{i\in M_{\text{InP}}} S_i||\sum_{n\in N_{\text{MVNO}}} U_n|$ do
    for $u \in U_n$ do
        Initialize $k' = 0, a_{u,j}^{k'} = a_{u,j}^{(0)}, \lambda_{n,u,j}^{k'}, a_{n,u,j}^{k'}$. Then, $a_{n,u,j}^{k'}$ is assigned by (54).
        while $\| \sum_{k=1}^{N}\sum_{u\in U_k}\sum_{i=1}^{M}\sum_{j\in S_i}(a_{n,u,z}^{k'} - a_{u,z}^{k'}) \|_2 > \varepsilon_0$ do
            At each MVNO n, updating $a_{n,u,z}^{k'+1}$ by (34), shown at the bottom of the next page. The data center collects $a_{n,u,z}^{k'+1}$ from each MVNO, then updates $a_{u,z}^{k'+1}$ by (32), and broadcasts them to all MNVOs. At each MVNO n, updating $\lambda_{n,u,z}^{k'+1}$ by (35), $k' = k' + 1$.
        end while
        Update $L \leftarrow L + \{L'_\rho(a_{n,u,z}^{k'})\}$.
    end for
end for
for $j = 1 : |\sum_{i\in M_{\text{InP}}} S_i|$ do
    if $j == \arg\min_{z\in S_i, i\in M_{\text{InP}}} L$ then
        $a_{u,j}^{(k+1)} \leftarrow a_{u,z}^{k'}$.
    else
        $a_{u,j}^{(k+1)} \leftarrow 0$.
    end if
end for
Output: $a_{u,j}^{(k+1)}, j \in S_i, i \in M_{\text{InP}}$.
```

The updating process of $a_{u,j}^{(k+1)}$ is presented as *Algorithm 1*. The iteration stop criterion is $\|\sum_{k=1}^{N}\sum_{u\in U_k}\sum_{i=1}^{M}\sum_{j\in S_i}(a_{n,u,j}^{k'} - a_{u,j}^{k'})\|_2 \le \varepsilon_0$, in which ε_0 is the tolerance for the primal and dual feasibility conditions in ADMM [15]. Note that in each iteration of Algorithm 1, each MVNO $n \in N_{\text{MVNO}}$ updates $a_{n,u,z}^{(k'+1)}$ and $\lambda_{n,u,z}^{k'+1}$ independently, while a data center is employed to collect and exchange data for them.

b) *Updating $y_{u,j}^{(k+1)}$:* When $a_{u,j}^{(k+1)}$ is obtained, $y_{u,j}^{(k+1)}$ is updated by solving the following problem.

$$\min_{y_{u,j}} \quad L_\rho\Big(y_{u,j},\ a_{u,j}^{(k+1)},\ \lambda_{u,j}^{(k)}\Big) = \sum_{k=1}^{N}\sum_{u\in U_k}\sum_{i=1}^{M}\sum_{j\in S_i}\Big[\frac{\rho}{2}\Big(a_{u,j}^{(k+1)} - y_{u,j}\Big)^2 + P y_{u,j} + \lambda_{u,j}^{k}\Big(a_{u,j}^{(k+1)} - y_{u,j}\Big)\Big] \quad (35)$$
$$\text{s.t.} \quad y_{u,j} \in \Phi_1, \quad \forall u, j.$$

Similarly, note that set Φ_1 in (19) and the objective function (35) are decomposable for each InP $i \in M_{\text{InP}}$. Thus, it is decomposed to multiple subproblems, which is solved by each InP individually. That is, for each SBS and MEC $j \in S_i$, InP *i* solves the following problem.

$$\min_{y_{u,j}} \quad L'_\rho\Big(y_{u,j},\ a_{u,j}^{(k+1)},\ \lambda_{u,j}^{(k)}\Big) = \sum_{n=1}^{N}\sum_{u\in U_n}\Big[\frac{\rho}{2}y_{u,j}^2 + \Big(P_u - \lambda_{u,j}^{(k)} - \rho a_{u,j}^{(k+1)}\Big)y_{u,j}\Big] \quad (36)$$

$$\text{s.t.} \quad y_{u,j} \geq 0, \ \forall u \in U_n, \forall n \in N_{\text{MVNO}}, \ \ \forall j \in S_i, \tag{37}$$

$$\sum_{n=1}^{N} \sum_{u \in U_n} y_{u,j} \leq t, \quad \forall j \in S_i. \tag{38}$$

Objective function (36) is obtained by decomposing (35) into *M* subproblems and removing the terms that do not contain variable $y_{u,j}$. Constraints (37) and (38) come from the definition of set Φ_1 in (19). To solve problem (36) efficiently, we propose a proposition to give out its optimal solution by the closed-form expression.

Proposition 3: For each $j \in S_i$, the optimal solution of problem (36) are given in (39), shown at the bottom of this page, in which $\gamma = -\frac{(tP_u + \sum_{u \in w}(P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)}))}{|w|}$, $w = \{u | P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)} < 0, u \in U_n, \forall n \in N_{\text{MVNO}}\}$.

Proof: It can be observed that objective function (36) is the sum of $|\sum_{k \in N_{\text{MVNO}}} U_k|$ quadratic functions with respect to $y_{u,j}$, which is convex. Meanwhile, constraints (37) and (38) are all linear constraints, which are also convex. Thus, the augmented lagrangian function of problem (36) can be expressed as

$$L(\mathbf{y}, \boldsymbol{\beta}) = \sum_{k=1}^{N} \sum_{u \in U_k} \left[\frac{\rho}{2} y_{u,j}^2 + \left(P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)} \right) y_{u,j} \right] + \beta_1 \left(\sum_{k=1}^{N} \sum_{u \in U_k} y_{u,j} - t \right) - \sum_{k=1}^{N} \sum_{u \in U_k} \beta_u y_{u,j}, \tag{40}$$

where β_1 and β_u are the Lagrange multipliers of constraints (38) and (37), respectively. By utilizing K. K. T. condition, the corresponding equation set (41)-(47) shown at the bottom of the next page, can be founded at the top of next page. This equation set is solved by the following discussion and derivation. Firstly, when $P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)} > 0$, since the terms $\rho y_{u,j}^*$ and β_1 are all non-negative according to (41) and (48), it can be derived that $\beta_u > 0$ from (41). By substituting $\beta_u > 0$ into (44), it indicates that $y_{u,j}^* = 0$, which is the first case of $y_{u,j}$ in (39). Secondly, when $P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)} < 0$, we assume $y_{u,j}^* = 0$. By substituting these into (41), it is obtained that $\beta_1 > 0$. By substituting $\beta_1 > 0$ into (44), there is $\sum_{k=1}^{N} \sum_{u \in U_k} y_{u,j}^* - t = 0$. This is contradicted with the assumption that $y_{u,j}^* = 0$, because in that case $\sum_{k=1}^{N} \sum_{u \in U_k} y_{u,j}^* = 0$. Thus, when $P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)} < 0$, there is $y_{u,j}^* > 0$. By substituting $y_{u,j}^* > 0$ into (45), it can be obtained that $\beta_u = 0$. Next, by substituting $\beta_u = 0$ into (41), $y_{u,j}^*$ can be expressed as

$$y_{u,j}^* = \frac{\rho a_{u,j}^{(k+1)} + \lambda_{u,j}^{(k)} - P_u - \beta_1}{\rho}. \tag{48}$$

On the other hand, by utilizing the derived first case of (39), (46) can be rewritten as

$$\sum_{u \in w} y_{u,j}^* - t \leq 0, \tag{49}$$

in which set *w* is defined in Proposition 3. By substituting (48) into (49), it indicates that when $\frac{\sum_{u \in w}(\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u)}{\rho} > t$ holds, β_1 can be expressed as

$$\beta_1 = -\frac{t\rho + \sum_{u \in w} \left(P_u - \lambda_{u,j}^{(k)} - \rho a_{u,j}^{(k+1)} \right)}{|w|}. \tag{50}$$

By substituting (50) into (48), the second case of $y_{u,j}^*$ in (39) is derived.

When $\frac{\sum_{u \in w}(\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u)}{\rho} < t$, together with (42), there is

$$\frac{\sum_{u \in w} \left(\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u - \beta_1 \right)}{\rho} < t. \tag{51}$$

By substituting (51) into (44), it can be derived that $\beta_1 = 0$. Thus, the third case of (39) is obtained by substituting this result into (50).

$$a_{n,u,z}^* = \begin{cases} \min\left\{t, \frac{R_u}{r_{u,z}}\right\}, & \begin{cases} \text{if } \frac{R_T}{r_{u,z}} > \min\left\{t, \frac{R_u}{r_{u,z}}\right\} \ \cap \ b < -\frac{R_T}{r_{u,z}} \\ \text{or } \frac{R_T}{r_{u,z}} \leq \min\left\{t, \frac{R_u}{r_{u,z}}\right\} \ \cap \ b < -\min\left\{t, \frac{R_u}{r_{u,z}}\right\} \end{cases} \\ \frac{R_T}{r_{u,z}}, & \begin{cases} \text{if } \frac{R_T}{r_{u,z}} > \min\left\{t, \frac{R_u}{r_{u,z}}\right\} \ \cap \ -\frac{R_T}{r_{u,z}} \leq b \leq -\min\left\{t, \frac{R_u}{r_{u,z}}\right\} \\ \text{or } \frac{R_T}{r_{u,z}} \leq \min\left\{t, \frac{R_u}{r_{u,z}}\right\} \ \cap \ -\frac{R_T}{r_{u,z}} \leq b < 0 \end{cases} \\ -b, & \begin{cases} \text{if } \frac{R_T}{r_{u,z}} > \min\left\{t, \frac{R_u}{r_{u,z}}\right\} \ \cap \ -\min\left\{t, \frac{R_u}{r_{u,z}}\right\} < b < 0 \\ \text{or } \frac{R_T}{r_{u,z}} \leq \min\left\{t, \frac{R_u}{r_{u,z}}\right\} \ \cap \ -\min\left\{t, \frac{R_u}{r_{u,z}}\right\} \leq b - \frac{R_T}{r_{u,z}} \end{cases} \\ 0, \quad b \geq 0 \end{cases} \tag{34}$$

$$y_{u,j}^* = \begin{cases} 0, \ \text{if } \ P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)} \geq 0 \\ \frac{\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u - \gamma}{\rho}, \ \text{if } \ P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)} < 0 \quad \&\& \quad \frac{\sum_{u \in w}\left(\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u\right)}{\rho} > t \\ \frac{\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u}{\rho}, \ \text{if } \ P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)} < 0 \quad \&\& \quad \frac{\sum_{u \in w}\left(\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u\right)}{\rho} \leq t \end{cases} \tag{39}$$

Algorithm 2 For Updating $y_{u,j}^{(k+1)}$

```
Input: $\lambda_{u,j}^{(k)}, a_{u,j}^{(k+1)}, P_u, t, \rho, \gamma = 0, w = \emptyset, u \in U_n, n \in N_{\mathrm{MVNO}}, j \in S_i$.
Iteration $k + 1$:
for $u = 1{:}|\sum_{m \in N_{\mathrm{MVNO}}} U_m|$ do
    if $P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{k} \geq 0$ then
        $y_{u,j}^{(k+1)} = 0$.
    else
        Update $w \leftarrow w + \{u\}$.
    end if
end for
$\gamma \leftarrow -\frac{(tP_u + \sum_{u \in w}(P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)}))}{|w|}$.
for $u = 1{:}|w|$ do
    if $\sum_{u \in w}(\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u) \leq \rho t$ then
        $y_{u,j}^{(k+1)} \leftarrow \frac{\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u}{\rho}$.
    else
        $y_{u,j}^{(k+1)} \leftarrow \frac{\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u - \gamma}{\rho}$.
    end if
end for
Output: $y_{u,j}^{(k+1)}, u \in U_n, n \in N_{\mathrm{MVNO}}$.
```

When $\frac{\sum_{u \in w}(\lambda_{u,j}^{(k)} + \rho a_{u,j}^{(k+1)} - P_u)}{\rho} = t$ holds, by utilizing simultaneous equation (48) and (49), it also can be derived that $\beta_1 = 0$, and this case is also included in the third case of (39). Moreover, when $P_u - \rho a_{u,j}^{(k+1)} - \lambda_{u,j}^{(k)} = 0$, we assume that $y_{u,j}^* \neq 0$ and substitute it into (45), there is $\beta_u = 0$. Then, (41) can be simplified as $\beta_1 = -\rho y_{u,j}^* < 0$, which is contradicted with (42). Hence, in this case, $y_{u,j}^* = 0$. We add this result into the first case of (39). Thus, Proposition 3 is proved. ■

The updating process of $y_{u,j}^{(k+1)}$ is presented as **Algorithm 2**, by which each InP $i \in M_{\mathrm{InP}}$ can update $y_{u,j}^{(k+1)}$ for its own SBSs and MEC servers.

c) *Updating* $\lambda_{u,j}^{(k+1)}$*:* When $a_{u,j}^{(k+1)}$ and $y_{u,j}^{(k+1)}$ are obtained, according to the procedure of ADMM [15], $\lambda_{u,j}^{(k+1)}$ can be computed as

$$\lambda_{u,j}^{(k+1)} = \lambda_{u,j}^{(k)} + \rho\Big(a_{u,j}^{(k+1)} - y_{u,j}^{(k+1)}\Big). \tag{52}$$

C. UE Power Optimization Under Fixed Offloading Association

For any fixed feasible offloading association ($\boldsymbol{a}, \boldsymbol{y}$), problem P2 degenerates into

$$\begin{aligned} \min_{P_u} \quad & \sum_{n=1}^{N} \sum_{u \in U_n} \left[P_u a_u^* + P_c \Big(R_u - B\log_2\Big(1 + \frac{P_u h_{u,j}^*}{N_0 + I}\Big) a_u^* \Big) \right] \\ \text{s.t.} \quad & 0 \leq P_u \leq P_{\max}, \forall u \in U_n, \forall n \in N_{\mathrm{MVNO}}. \end{aligned}$$

where $a_u^* = \sum_{i=1}^{M} \sum_{j \in S_i} a_{u,j}^* x_{u,j}^*$. $h_{u,j}^*$ is the channel gain between UE *u* and its associated SBS. Note that the objective function and the constraint are convex and decomposable for each $u \in U_n$. So each UE can optimize its power individually by K. K. T. condition, which can be expressed as

$$P_u^* = \begin{cases} \min\Big\{\frac{P_c B}{\ln 2} - \frac{N_0 + I}{h_{u,j}^*}, P_{\max}\Big\}, & \text{if } \frac{P_c B}{\ln 2} - \frac{N_0 + I}{h_{u,j}^*} \geq 0, \\ 0, & \text{if } \frac{P_c B}{\ln 2} - \frac{N_0 + I}{h_{u,j}^*} < 0. \end{cases} \tag{53}$$

The closed-form (53) provides some insights for the system design. Firstly, when P_c is large, P_u^* will become large to minimize the total energy consumption, because in this case, energy consumption is serious by local computing. Secondly, when $h_{u,j}^*$ is small, P_u^* will become small to avoid energy waste by bad channel.

Thus, the flow diagram of the proposed offloading algorithm is described as **Algorithm 3**, where the iteration stoping criterion in Step 6 is $\|\mathbf{a}^{(k+1)} - \mathbf{y}^{(k+1)}\|_2 \leq \varepsilon_1 \&\& \|\mathbf{a}^{(k+1)} - \mathbf{a}^{(k)}\|_2 \leq \varepsilon_2$, in which ε_1 and ε_2 are the tolerances for the primal and dual feasibility conditions in ADMM [15]. Due to the distributed manner, the overhead expense of the proposed algorithm is slight. Firstly, according to the expression of (39), the updating process of InPs in Step 4 only depends on the iteration result from MVNOs, while it has no correlation to any UEs, so the channel state information (CSI) collection is avoided. Moreover, although the updating process in Step 2 requires CSI collection, the expense is still manageable, because for each MVNO, it only collects the CSIs from its own subscribers.

$$\begin{cases} \frac{\partial L(\mathbf{y}, \boldsymbol{\beta})}{\partial y_{u,j}} = \rho(y_{u,j}^* - a_{u,j}^{(k+1)}) + P_u - \lambda_{u,j}^{(k)} + \beta_1 - \beta_u = 0, \\ \qquad \forall u \in U_n, \forall n \in N_{\mathrm{MVNO}} & (41) \\ \beta_1 \geq 0 & (42) \\ \beta_u \geq 0, \quad \forall u \in U_n, \forall n \in N_{\mathrm{MVNO}} & (43) \\ \beta_1(\sum_{n=1}^{N} \sum_{u \in U_n} y_{u,j}^* - t) = 0 & (44) \\ \beta_u y_{u,j}^* = 0, \quad \forall u \in U_n, \forall n \in N_{\mathrm{MVNO}} & (45) \\ \sum_{n=1}^{N} \sum_{u \in U_n} y_{u,j}^* - t \leq 0 & (46) \\ y_{u,j}^* \geq 0, \quad \forall u \in U_n, \forall n \in N_{\mathrm{MVNO}} & (47) \end{cases}$$

Algorithm 3 Distributed Algorithm for Offloading and Power Optimization.

```
Initialization: $k = 0, P_c, P_u, t, \rho, h_{u,j}, y_{u,j}^{(k)}, a_{u,j}^{(k)}, \lambda_{u,j}^{(k)}$.
Step1: Each MVNO $n$ collects $P_u, h_{u,j}, u \in U_n, t, \rho$. Each InP $i$ collects $P$, $t$, $\rho$. Both of them update $y_{u,j}^{(k)}$, $a_{u,j}^{(k)}$.
Step2: Each MVNO $n$ and the data run Algorithm 1 to update $a_{u,j}^{(k+1)}, u \in U_n$.
Step3: Each MVNO $n$ transmits $a_{u,j}^{(k+1)}$ to the involved InPs.
Step4: Each InP $i$ utilizes Algorithm 2 and the received $a_{u,j}^{(k+1)}$ to update $y_{u,j}^{(k+1)}, j \in S_i$. Then $\lambda_{u,j}^{(k+1)}$ is update by (52).
Step5: Each InP $i$ transmits $y_{u,j}^{(k+1)}, \lambda_{u,j}^{(k+1)}, j \in S_i$ to the involved MVNOs.
Step6: Each MVNO $n$ updates $P_u$ by (53). Then, the updated data is examined by the iteration stoping criterion. If it is satisfied, the iteration stops, and $a_{u,j}^{(k+1)}$ is allocated to the UEs for their association and offloading. Otherwise, $k \leftarrow k + 1$, and the procedure returns to Step 2.
```

D. Complexity Analysis

The complexity of Algorithm 3 is mainly caused by the updating of $a_{u,j}^{(k)}, y_{u,j}^{(k)}, \lambda_{u,j}^{(k)}$ and $P_u^{(k)}$ in each iteration. Since the computation of $\lambda_{u,j}^{(k)}$ and $P_u^{(k)}$ depends on the closed-form expressions (52) and (53), the complexity mainly comes from Algorithm 1 in step 2 and Algorithm 2 in step 4. For Algorithm 1, the complexity consists of two parts, one is traversal search on set $\{a_{n,u,z}\}$ for the maximum in (33), while the other one is the computation by (34) to obtain each element of set $\{a_{n,u,z}\}$. For the first part, by taking bubble sort algorithm as an example, the complexity is proportional to the size of set $\{a_{n,u,z}\}$, which is $\mathcal{O}(|\sum_{n\in N_{\mathrm{MVNO}}} U_n|^2|\sum_{i\in M_{\mathrm{InP}}} S_i|^2)$. For the second part, since (34) is in closed-form, the complexity is $\mathcal{O}(1)$, thus, the complexity of Algorithm 1 for all the UEs can be expressed as $|\sum_{n\in N_{\mathrm{MVNO}}} U_n|\mathcal{O}(|\sum_{n\in N_{\mathrm{MVNO}}} U_n|^2|\sum_{i\in M_{\mathrm{InP}}} S_i|^2)$. For Algorithm 2, the complexity in each iteration is mainly caused by two loops. The first loop is to construct set *w*, and its complexity is $\mathcal{O}(|\sum_{n\in N_{\mathrm{MVNO}}} U_n|)$. The second loop is to compute the non-zero element of set *w* with the closed-form (39), and the complexity is $\mathcal{O}(|w|)$. Since $|w| \leq |\sum_{n\in N_{\mathrm{MVNO}}} U_n|$, the complexity of Algorithm 2 for all MEC servers can be bounded as $|\sum_{i\in M_{\mathrm{InP}}} S_i|[\mathcal{O}(|w|)+\mathcal{O}(|\sum_{n\in N_{\mathrm{MVNO}}} U_n|)] \leq 2|\sum_{i\in M_{\mathrm{InP}}} S_i|\mathcal{O}(|\sum_{n\in N_{\mathrm{MVNO}}} U_n|)$. Finally, the complexity of step 6 for all UEs to optimize power P_u is $|\sum_{i\in M_{\mathrm{InP}}} S_i||\sum_{n\in N_{\mathrm{MVNO}}} U_n|\mathcal{O}(|1|)$. Therefore, the complexity of the proposed algorithm can be approximately expressed as $\mathcal{O}(|\sum_{n\in N_{\mathrm{MVNO}}} U_n|^3|\sum_{i\in M_{\mathrm{InP}}} S_i|^2)$, which is the sum of the mentioned parts. Comparing with the original problems P1 and P2, which have the complexity of $\mathcal{O}(2^{|\sum_{n\in N_{\mathrm{MVNO}}} U_n||\sum_{i\in M_{\mathrm{InP}}} S_i|})$ due to the integer variable constrains (7) and (11), it can be observed that the proposed decompositions and derivations are effective to reduce the complexity.

V. Simulation Results

In this section, we evaluate the proposed algorithm through extensive simulation experiments. The results are divided into three parts: **(i)** We compare the proposed algorithm with several benchmark algorithms in terms of energy consumption per bit, which is defined as the ratio of overall energy consumption of all UEs and data amount of all tasks. The impacts of the number of MEC servers and UEs on the evaluated algorithms are also studied and analyzed. **(ii)** The convergence of the proposed algorithm is studied, in which the impacts of penalty factor ρ are investigated. **(iii)** The complexity of the proposed algorithm is evaluated in term of running time on the simulation platform. In the simulations, there are two InPs coexisting in an area of 500 × 500 m^2. Each InP owns one MBS and 10 SBSs. Each SBS is integrated with one MEC server. There are three MVNOs coexisting in the area, each of which has 20 subscription UEs. All the UEs and SBSs follow normal distribution in the area. The spectrum bandwidth $B = 10$ MHz. The uplink duration $t = 50$. The power of the background noise $N_0 = 10^{-9}$ W, while the interference threshold $I = 1.7 \times 10^{-9}$ W. For each UE u, the data amount of the task $R_u = 10$ M bits. The range of the transmission power P_u is [0.6, 1.5] W. The energy consumption of local computing $P_c = 2 \times 10^{-8}$ J/bit. The overall throughput threshold $R_T = 80$ M bits. The penalty factor $\rho_s = 0.01$ for the loop in Algorithm 1. In the beginning of each iteration, the initial values $y_{u,j}^{(0)}, a_{u,j}^{(0)}$ and $a_{n,u,j}^{0}$ are randomly selected from 20 to 50. $\lambda_{u,j}^{(0)}$ and $\lambda_{u,j}^{0}$ are randomly selected from 1 to 2. $P_u^{(0)}$ is randomly selected from 1 to 1.5. The channel gain between each UE and SBS is modeled as the norm of zero-mean, independent, circular-symmetric complex Gaussian random variable with variances 1 while $\varphi = 1.9$. The iteration stoping thresholds $\varepsilon_0 = \varepsilon_1 = 0.01$ and $\varepsilon_2 = 0.001$.

All the simulations run on a MATLAB-based platform, in which the CPU is Intel i5-4210U and the capacity of the RAM is 8 GB. For fair evaluation, three baseline algorithms are also simulated and compared. The fist one is the optimal solution, which is obtained by inputting problem P1 into CPLEX 12.5 [30]. Here, CPLEX is employed through YALMIP tool [31] in MATLAB environment. The second one is BP algorithm in [28] by setting $\alpha = 0, \beta = 1$ in its objective function. Moreover, to use the BP algorithm in this comparison, it is run at each MEC server while each MEC server only includes 3 UEs that have superior channel gains, so that the integer variables can be eliminated. The last one is the centralized solution in the convergence study, which is to optimize the two variables of problem P3 jointly by the dual ascent method [15]. For all the benchmarks, the transmission power is fixed to 1 W.

Fig. 2 illustrates the energy consumptions of the compared algorithms with respect to the number of iteration, in which each MVNO owns 20 UEs while each InP controls 10 SBSs. With various values of ρ, the proposed algorithm converges to the same value of energy consumption, which is only within 10% larger than the optimal solution. Due to the existence of the residuals and tolerances for the primal and dual feasibility condition [15], the proposed algorithm is always

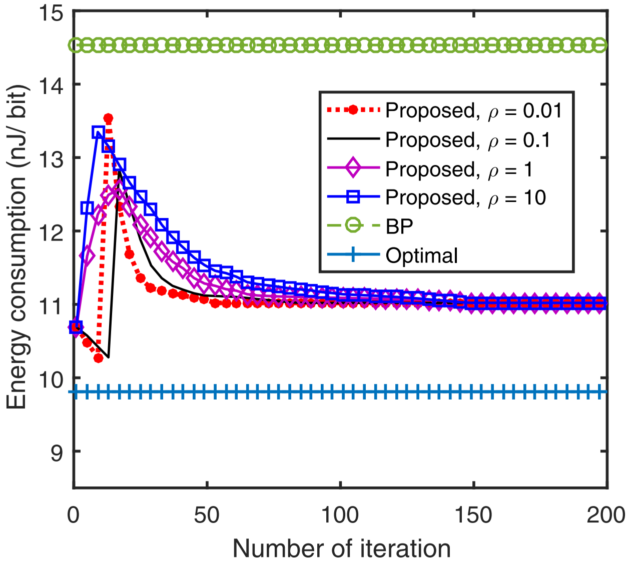

Fig. 2. Energy consumptions of the compared algorithms with $\{|U_n| = 20, |S_j| = 10\}$.

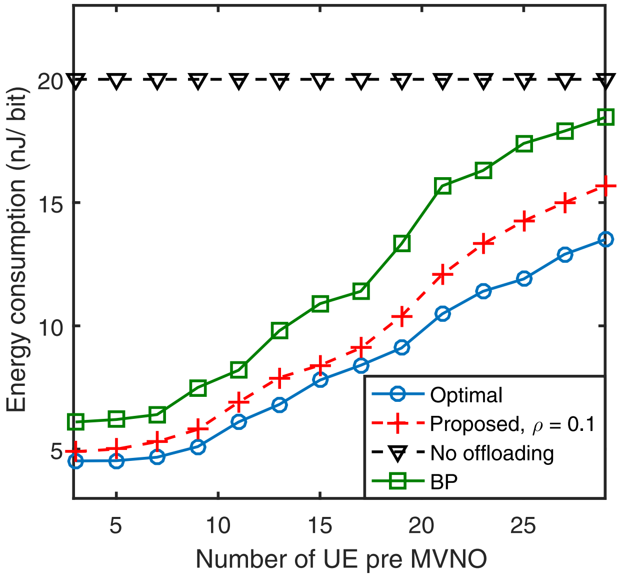

Fig. 3. Energy consumptions of the compared algorithms versus the number of UEs pre MVNO.

inferior to the optimal solution, but the narrow gap shows its effectiveness in reducing the energy consumption of the UEs. In addition, although the proposed algorithm can converges to the same value with different ρ, the convergence speeds varies. When $\rho = 0.01$, the proposed algorithm converges with almost 60 iterations, but when $\rho = 10$, it takes 160 iterations to converge, which indicates that the proposed algorithm takes less number of iterations with small values of ρ. Finally, the figure shows that the proposed algorithm outperforms the BP algorithm. Although both the two compared algorithms solve the problem in a distributed manner, some performance losses are introduced into the BP-based algorithm when utilizing greedy method to handle the integer variables. Different from that, through problem transformation and decomposition, the proposed algorithm solves the integer variables by traversal search in a small set, so the desirable tradeoff between performance and complexity can be achieved.

Then we evaluate the impacts of the number of UE on energy consumption. As shown in Fig. 3, the number of UE pre MVNO increases from 2 to 29, while the number of MEC servers pre InP is fixed as 10. For the proposed algorithm, $\rho = 0.1$ and the number of iteration is 90. We also simulate the metric of no offloading as the baseline. In general, all the compared schemes outperform the one of no offloading, which verifies the positive effect of offloading in energy reduction. The proposed algorithm is superior to BP scheme, while it is also close to the optimal solution. Besides, when the number of UE pre MVNO is small, all the compared curves tend to be flat as the number of UE increases. In this case, since the access capacity of the MECs far exceeds the overall UE rates, all the UEs offload their total task volume to the MECs driven by throughput constraint (6) in the proposed algorithm, so the energy consumption per bit mainly depends on data transmission. On the other hand, when the number of UE pre MVNO exceeds 11 and keeps increasing, the metrics of all the compared schemes also increase. In the case, restrained by uplink duration, the overall UE rates exceeds the access capacity of the MECs, so more and more tasks are handled by local computing other than offloaded to MECs, which leads to higher energy consumption per bit.

Another performance about effect of the number of MEC servers on energy consumption is revealed in Fig. 4. In this comparison, the number of MEC servers pre InP increases from 1 to 24, while the one of UEs pre MVNO is fixed as 20. As shown in the figure, when the number of MEC servers pre InP is small, the metrics of all compared algorithms are close to that of no offloading. In fact, when there are few MEC servers in the networks, the advantage of offloading is restricted by servers capacity, and the vast majority of the tasks are finished by local computing, which leads to high energy consumption. As the number of MEC servers pre InP increases, energy consumption per bit keeps decline, while the gaps of the compared algorithms become obvious. The proposed algorithm is superior to BP scheme, and it is close to the optimal solution. In this case, servers capacity increases with the number of servers. More and more tasks are offloaded, so the proportion of tasks performed by local computing is becoming smaller and smaller, resulting in a decrease in energy consumption. Moreover, when the number of MEC servers exceeds 20, energy consumption no longer decreases. At this point, the servers capacity has been able to fully accommodate all UEs, so the energy consumption per bit mainly depends on data transmission. This result indicates that when the network is configured with enough MEC servers, it is helpless to reduce energy consumption by adding more MEC servers.

In Fig. 5, we compare the cumulative distribution probability (CDF) of the proposed algorithm with various setting of penalty factor ρ. The centralized solution is also simulated as baseline. As depicted in Fig. 5, the proposed algorithm outperforms the baseline in convergence speed. Comparing with the centralized solution, the proposed algorithm only updates

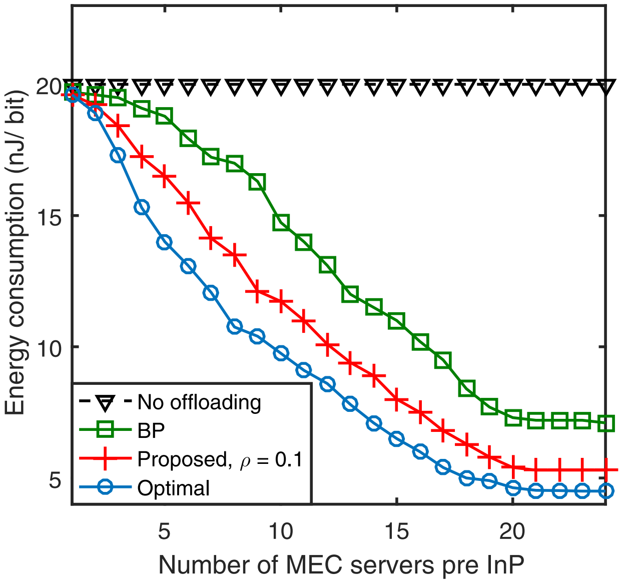

Fig. 4. Energy consumptions of the compared algorithms versus the number of MVNOs pre InP.

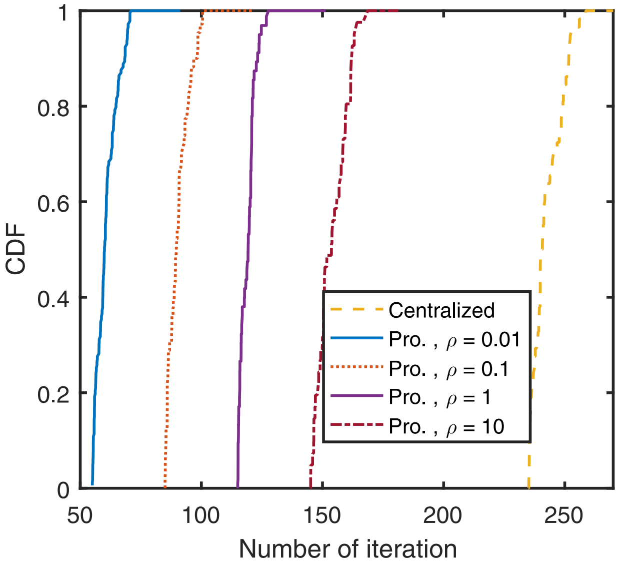

Fig. 5. Cumulative distribution probability of the proposed algorithm with various penalty factor ρ.

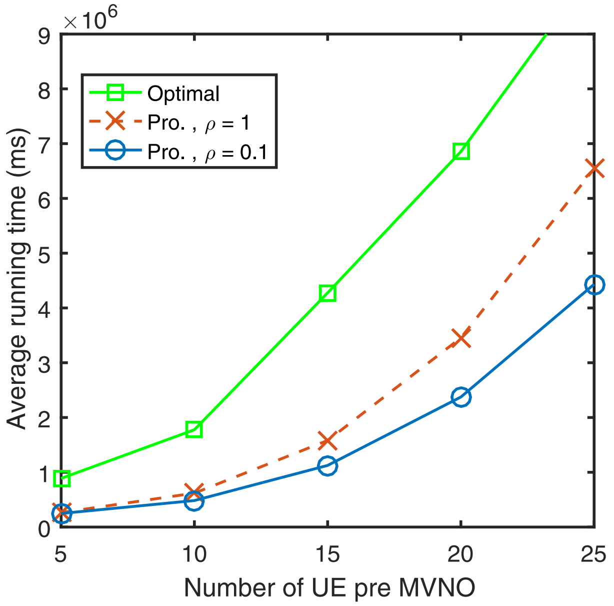

Fig. 6. Average running time of the compared algorithms verses the number of UEs pre MVNO.

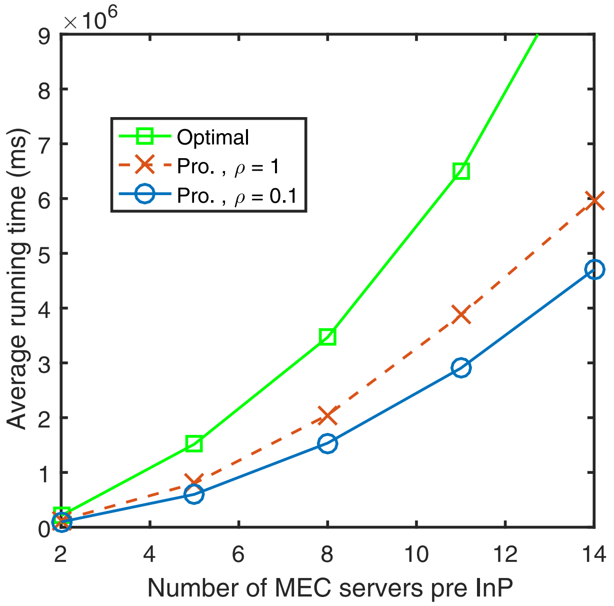

Fig. 7. Average running time of the compared algorithms verses the number of MEC servers pre InP.

one variable in one iteration mainly depending on the derived closed-form expression (34) and (39), so the proposed algorithm has lower complexity and converge faster. Furthermore, it indicates that when ρ is smaller, the proposed algorithm converges with less iterations. The trend is consistent with the result in Fig. 2. From these experiments, it is also verified that the order of magnitudes of ρ should approach the one of objective function to achieve desirable convergence.

Next, we examine the complexity of the proposed algorithm by average running time, which is measured by MATLAB commands *tic* and *toc*. We also utilize CPLEX 12.5 to solve problem P1 in MATLAB environment through YAMLP tool, which is compared as the baseline. Fig. 6 depicts the running time of the compared algorithms verses the number of UEs pre MVNO. In the simulation, the number of MEC servers pre InP is fixed to 10. It can be seen that the running time of the baseline increases exponentially as the number of UEs grows. This is caused by the integer variable $x_{u,j}$ in P1, the number of which equals the one of the UEs. It indicates that the complexity is mainly caused by the integer variables. Comparing with the baseline, the metric of the proposed algorithm increases slowly as the number of UEs grows. The result corresponds with the complexity analysis in Section IV, since the complexity of the proposed algorithm is cubic with respect to the number of UEs. Moreover, when $\rho = 0.1$, the proposed algorithm converges faster than that of 1. This is due to the number of iterations, because it requires less iterations when ρ is small. Besides, because of the derived closed-form expressions

(34) and (39), the complexity is largely reduced. Hence, when $\rho = 1$, the additional running time caused by the iterations is not obvious comparing with the one when $\rho = 0.1$.

Another comparison about the relationship between complexity and the number of MEC servers is illustrated in Fig. 7. Here, the number of UEs pre MVNO is fixed as 20. Similarly, the metric of the optimal solution increases exponentially as the number of MEC servers grows. In fact, although the number of UEs is fixed, the combinations also increases as MEC servers grows, which is of high complexity. The proposed algorithm outperforms the baseline, while the slope of the time growth is smaller than that in Fig. 6. This result corresponds well with the complexity analysis in Section IV, because the complexity of the proposed algorithm is quadratic with respect to the number of MEC servers. Combining the results in Figs. 6 and 7, it verifies the effectiveness of the proposed decomposition and derivation in reducing the complexity of the problem.

VI. Conclusion

In this paper, we investigated the offloading problem for virtualized SCNs with MEC, which aims to minimize the energy consumption of all the UEs subject to minimum overall throughput of the network. The problem was formulated as a MINLP, which considered jointly offloading, time slice and power allocation among multiple UEs, MVNOs and InPs. To reduce the complexity, a distributed framework was developed, where the original problem was converted into a biconvex problem and decomposed into two subproblems. Towards the first subproblem, local variables were introduced to handle the coupling constraint, so that the problem was transformed into the desired form. Then an alternating direction method of multipliers (ADMM)- based algorithm is proposed to solve it efficiently. To reduce the complexity further, we derived the closed-form expressions of the optimal solutions for the variables updating of the both subproblems. Detailed simulations verified that the proposed algorithm can efficiently handle the network resource allocation and energy consumption minimization of the UEs with manageable complexity. Besides, the effectiveness of the proposed decomposition and derivation in complexity reduction were also confirmed. In future work, we will consider virtualization and MEC paradigm in Internet of Things to improve the energy efficiency.

Appendix A
Proof of Lemma 1

For variable $y_{u,j}$, by substituting $P_u = \tilde{P_u}$ into (15), it can be simplified to the sum of series of linear function $\tilde{P_u} y_{u,j}$, which is convex in $y_{u,j}$. Besides, the involved feasible set Φ_1 is also convex, since it is the union of two linear constraint of $y_{u,j}$ by (19). Similarly, for variable $a_{u,j}$, (15) can be simplified as the sum of series of linear function $-r_{u,j}(\tilde{P_u}) a_{u,j}$, while the involved feasible set Φ_2 is also convex in $a_{u,j}$ by (20), therefore, the Lemma is proved.

Appendix B
Proof of Lemma 2

Similar to Lemma 1, for variable P_u, (15) can be simplified as the sum of series of linear function $\tilde{y}_{u,j} P_u$, while the involved feasible set Φ_3 is affine. Therefore, the Lemma is proved.

Appendix B
Proof of Lemma 3

For variable P_u, (15) can be simplified as a affine function with terms $-B\log_2(1+\frac{P_u h_{u,j}}{N_0+I})\tilde{a}_{u,j}$. Since the log function of P_u is concave, the negative one is convex. Hence, $f_2(\tilde{a}_{u,j}, P_u)$ is convex in P_u. Combining with the affine set Φ_3, the Lemma is proved.

Appendix C
Proof of Proposition 2

According to constraint (11), it can be deduced that for each $u \in U_k$, the vector $[x_{u,1}, \ldots, x_{u,j}, \ldots, x_{u,|\sum_{i \in M_{\text{InP}}} S_i|}] \in D$, in which $D = \{\mathbf{x}_z | \mathbf{x}_z = [0, \ldots, x_z, 0, \ldots,], x_z = 1, z = 1, 2, \ldots, |\sum_{i \in M_{\text{InP}}} S_i|\}$. We utilize traversal search in set D to fix binary variable $x_{u,j}$, that is, for each $\mathbf{x}_z \in D$, we set

$$x_{u,j} = \begin{cases} 0, & j \neq z, \\ 1, & j = z. \end{cases} \quad \text{and} \quad a_{n,u,j} = \begin{cases} 0, & j \neq z, \\ a_{n,u,z}, & j = z. \end{cases} \tag{54}$$

After that, by substituting $x_{u,j}$ and $a_{u,j}$ back to problem (25), it can be simplified as

$$\begin{aligned} \min_{a_{n,u,z}} \quad & L'_\rho\Big(a_{n,u,z},\ y^{(k)}_{u,j},\ \lambda^{(k)}_{u,j}\Big) + \sum_{n=1}^{N} \sum_{u \in U_n} \sum_{i=1}^{M} \sum_{j \in S_i} \\ & \Big[\lambda_{n,u,j} \times \Big(a_{n,u,z} - a^{k'}_{u,j}\Big) + \frac{\rho_s}{2}\Big(a_{n,u,z} - a^{k'}_{u,j}\Big)^2\Big] \\ \text{s.t.} \quad & a_{n,u,z} \in \Phi_2, \quad \forall u, j. \end{aligned} \tag{55}$$

It can be observed that the objective function is convex, and the feasible set Φ_2 is convex set, so the above problem is convex. By solving the corresponding K. K. T. conditions, $a^*_{n,u,z}$ can be obtained as (38) for each $\mathbf{x}_z \in D$. After that, the set $\{L'_\rho(a^*_{n,u,z}) | z = 1, 2, \ldots, |\sum_{i \in M_{\text{InP}}} S_i|\}$ can be obtained. Find the minimum of $L'_\rho(a^*_{n,u,z})$ and set the corresponding argument $a^*_{n,u,z}$ as $a^*_{n,u,j}$, while set the others to 0. This process can be expressed as (37). Thus, the solution $a^{k'+1}_{n,u,j}$ for step (33) is obtained. By substituting $a^{k'+1}_{n,u,j}$ into (34) and setting the derivation to 0, $a^{k'+1}_{u,j}$ can be derived as

$$a^{k'+1}_{u,j} = \frac{\sum_{n=1}^{N} \sum_{u \in U_n} \sum_{i=1}^{M} \sum_{j \in S_i} \Big(a^{k'+1}_{n,u,j} + \frac{\lambda^{k'}_{n,u,j}}{\rho_s}\Big)}{|\sum_{u \in U_n} \sum_{n \in N_{\text{MVNO}}} U_n| |\sum_{i \in M_{\text{InP}}} S_i|}. \tag{56}$$

By substituting ADMM dual variable relationship [15] $\sum_{n=1}^{N} \sum_{u \in U_n} \sum_{i=1}^{M} \sum_{j \in S_i} \lambda^{k'}_{n,u,j} = 0$ into (60), $a^{k'+1}_{u,j}$ can be obtained like (36) in each iteration. When the iteration stops, by taking $a^*_{u,j} = a^{k'}_{u,j}$, (36) is obtained, and the proposition is proved.

References

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[2] J. A. Stankovic, "Research directions for the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.

[3] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[4] J. G. Andrews *et al.*, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.

[5] C.-X. Wang *et al.*, "Cellular architecture and key technologies for 5G wireless communication networks," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 122–130, Feb. 2014.

[6] V. Jungnickel *et al.*, "The role of small cells, coordinated multipoint, and massive MIMO in 5G," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 44–51, May 2014.

[7] O. Galinina, A. Pyattaev, S. Andreev, M. Dohler, and Y. Koucheryavy, "5G multi-RAT LTE-WiFi ultra-dense small cells: Performance dynamics, architecture, and trends," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 6, pp. 1224–1240, Mar. 2015.

[8] M. Satyanarayanan, "The emergence of edge computing," *IEEE Comput.*, vol. 50, no. 1, pp. 30–39, Jan. 2017.

[9] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[10] *Mobile-Edge Computing—Introductory Technical White Paper*, ETSI, Sophia Antipolis, France, Sep. 2014.

[11] C. Liang and F. R. Yu, "Wireless network virtualization: A survey, some research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 358–380, 1st Quart., 2015.

[12] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, "NVS: A substrate for virtualizing wireless resources in cellular networks," *IEEE/ACM Trans. Netw.*, vol. 20, no. 5, pp. 1333–1346, Oct. 2012.

[13] X. Costa-Perez, J. Swetina, T. Guo, R. Mahindra, and S. Rangarajan, "Radio access network virtualization for future mobile carrier networks," *IEEE Commun. Mag.*, vol. 51, no. 7, pp. 27–35, Jul. 2014.

[14] S. Singh, H. S. Dhillon, and J. G. Andrews, "Offloading in heterogeneous networks: Modeling, analysis, design insights," *IEEE Trans. Wireless Commun.*, vol. 12, no. 5, pp. 2484–2497, May 2013.

[15] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends® Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[16] Q. Chen, G. Yu, H. Shan, A. Maaref, G. Y. Li, and A. Huang, "Cellular meets WiFi: Traffic offloading or resource sharing?" *IEEE Trans. Wireless Commun.*, vol. 15, no. 5, pp. 3354–3367, May 2016.

[17] S. Zhang, N. Zhang, S. Zhou, J. Gong, Z. Niu, and X. Shen, "Energy-aware traffic offloading for green heterogeneous networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1116–1129, May 2016.

[18] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.

[19] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile edge computing," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[20] Y. Mao, J. Zhang, K. B. Letaief, "Dynamic computation offloading for mobile edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[21] C. You, K. Huang, H. Chae, and B.-H. Kim, "Energy-efficient resource allocation for mobile edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1397–1411, Mar. 2017.

[22] M. Chen and Y. Hao, "Task offloading for mobile edge computing in software defined ultra-dense network," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 587–597, Mar. 2018.

[23] H. Guo, J. Liu, J. Zhang, W. Sun, and N. Kato, "Mobile edge computation offloading for ultra-dense IoT networks," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4977–4988, Dec. 2018.

[24] Z. Tan, F. R. Yu, X. Li, H. Ji, and V. C. M. Leung, "Virtual resources allocation for heterogeneous service in full duplex-enabled small cell networks with cache and MEC," in *Proc. IEEE INFOCOM Workshops*, Atlanta, GA, USA, May 2017, pp. 163–168.

[25] Z. Tan, F. R. Yu, X. Li, H. Ji, and V. C. M. Leung, "Virtual resource allocation for heterogeneous services in full duplex-enabled SCNs with mobile edge computing and caching," *IEEE Trans. Veh. Commun.*, vol. 67, no. 2, pp. 1794–1808, Feb. 2018.

[26] F. Guo, H. Zhang, H. Ji, X. Li, and V. C. M. Leung, "Energy efficient computation offloading for multi-access MEC enabled small cell networks," in *Proc. IEEE ICC Workshops*, Kansas City, MO, USA, May 2018, pp. 1–6.

[27] F. Guo, H. Zhang, H. Ji, X. Li, and V. C. M. Leung, "An efficient computation offloading management scheme in the densely deployed small cell networks with mobile edge computing," *IEEE/ACM Trans. Netw.*, vol. 26, no. 6, pp. 2651–2664, Oct. 2018.

[28] J. Li, A. Wu, S. Chu, T. Liu, and F. Shu, "Mobile edge computing for task offloading in small cell networks via belief propagation," in *Proc. IEEE ICC*, Kansas City, MO, USA, May 2018, pp. 1–6.

[29] S. Diamond, R. Takapoui, and S. P. Boyd, "A general system for heuristic minimization of convex functions over non-convex sets," *Optim. Methods Softw.*, vol. 33, no. 1, pp. 165–193, Feb. 2017.

[30] IBM. (2015). *ILOG CPLEX Optimizer*. [Online]. Available: http://www.ibm.com/

[31] J. Lofberg, "YALMIP: A toolbox for modeling and optimization in MATLAB," in *Proc. IEEE Int. Conf. Robot. Autom.*, New Orleans, LA, USA, Sep. 2004, pp. 284–289.

[32] L. Chen, F. R. Yu, H. Ji, G. Liu, and V. C. M. Leung, "Distributed virtual resource allocation in small-cell networks with full-duplex self-backhauls and virtualization," *IEEE Trans. Veh. Technol.*, vol. 65, no. 7, pp. 5410–5423, Jul. 2016.

[33] D. W. K. Ng and R. Schober, "Resource allocation and scheduling in multi-Cell OFDMA systems with decode-and-forward relaying," *IEEE Trans. Wireless Commun.*, vol. 10, no. 7, pp. 5410–5423, Jul. 2011.

[34] H. Zhang, C. Jiang, N. C. Beaulieu, X. Chu, X. Wen, and M. Tao, "Resource allocation in spectrum-sharing OFDMA femtocells with heterogeneous services," *IEEE Trans. Commun.*, verogeneous services in full duplex-enabled SCNs with mobile edge computing and caching," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1794–1808, Feb. 2018.

[35] J. Gorski, F. Pfeuffer, and K. Klamroth, "Biconvex sets and optimization with biconvex functions: A survey and extensions," *Math. Methods Oper. Res.*, vol. 66, no. 3, pp. 373–407, Dec. 2007.

Yulun Cheng was born in Wuyuan County, Inner Mongolia, in 1983. He received the B.E. degree in information and communication engineering and the Ph.D. degree in communication and information system from the College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, China, in 2007 and 2014, respectively, where he has been with the Faculty of the Jiangsu Key Laboratory of Wireless Communications since 2014. He is currently a Post-Doctoral Research Fellow with the China Information Consulting and Designing Institute Company Ltd. His research interests include wireless virtual networks, NFV, and Internet of Things.

Jun Zhang (S'10–M'14) received the M.S. degree in statistics from the Department of Mathematics, Southeast University, Nanjing, China, in 2009, and the Ph.D. degree in communications information system from the National Mobile Communications Research Laboratory, Southeast University, Nanjing, in 2013. From 2013 to 2015, he was a Post-Doctoral Research Fellow with the Singapore University of Technology and Design, Singapore. Since 2015, he has been with the Faculty of the Jiangsu Key Laboratory of Wireless Communications, College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, where he is currently an Associate Professor. His research interests include massive MIMO communications, physical layer security, edge caching and computing, and large dimensional random matrix theory. He was a recipient of the Globcom Best Paper Award in 2016 and the IEEE APCC Best Paper Award in 2017. He serves as an Associate Editor for the IEEE COMMUNICATIONS LETTERS.

Longxiang Yang is with the College of Communications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China, where he is a Full Professor and the Doctoral Supervisor. He has fulfilled multiple National Natural Science Foundation projects of China. He has authored and coauthored over 100 technical papers published in various journals and conferences. His research interests include cooperative communication and network coding, wireless communication theories, key technologies of mobile communication systems, ubiquitous networks, and Internet of Things.

Chenming Zhu is the Chief Technical Officer of China Information Consulting and Designing Institute Company Ltd., Nanjing, China, where he is a Post-Doctoral Tutor. He has authored and coauthored over 40 technical papers published in various journals and conferences. His research interests include mobile communication network planning, designing and optimization, key technologies of mobile communication systems, and Internet of Things.

Hongbo Zhu received the B.S. degree in communications engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, and the Ph.D. degree in information and communications engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 1982 and 1996, respectively. He is currently a Professor and the Vice-President with the Nanjing University of Posts and Telecommunications. He is also the Head of the Coordination Innovative Center, IoT Technology and Application, which is the first governmental authorized Coordination Innovative Center, IoT in China. He also serves as a referee or expert in multiple national organizations and committees. He has authored and coauthored over 200 technical papers published in various journals and conferences. He is currently leading a big group and multiple funds on IoT and wireless communications with current focus on architecture and enabling technologies for Internet of Things. His research interests include mobile communications, wireless communication theory, and electromagnetic compatibility.