

Article

Joint Clustering and Resource Allocation Optimization in Ultra-Dense Networks with Multiple Drones as Small Cells Using Game Theory [†]

Tinh T. Bui ^{1,2} , Long D. Nguyen ^{3,4,*} , Ha Hoang Kha ^{1,2} , Nguyen-Son Vo ^{3,4}  and Trung Q. Duong ^{5,*} 

¹ Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City 700000, Vietnam

² Vietnam National University Ho Chi Minh City, Linh Trung Ward, Thu Duc District, Ho Chi Minh City 700000, Vietnam

³ Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam

⁴ School of Engineering and Technology, Duy Tan University, Da Nang 550000, Vietnam

⁵ School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Belfast BT3 9DT, UK

* Correspondence: nguyendinhlong1@duytan.edu.vn (L.D.N.); trung.q.duong@qub.ac.uk (T.Q.D.)

[†] This paper is an extended version of our paper published in 2022 IEEE 9th International Conference on Communications and Electronics (ICCE), Nha Trang, Vietnam, 27–29 July 2022.

Abstract: In this study, we consider the combination of clustering and resource allocation based on game theory in ultra-dense networks that consist of multiple macrocells using massive multiple-input multiple-output and a vast number of randomly distributed drones serving as small-cell base stations. In particular, to mitigate the intercell interference, we propose a coalition game for clustering small cells, with the utility function being the ratio of signal to interference. Then, the optimization problem of resource allocation is divided into two subproblems: subchannel allocation and power allocation. We use the Hungarian method, which is efficient for solving binary optimization problems, to assign the subchannels to users in each cluster of small cells. Additionally, a centralized algorithm with low computational complexity and a distributed algorithm based on the Stackelberg game are provided to maximize the network energy efficiency (EE). The numerical results demonstrate that the game-based method outperforms the centralized method in terms of execution time in small cells and is better than traditional clustering in terms of EE.

Keywords: game theory; ultra-dense network; energy efficiency; unmanned aerial vehicle (UAV); massive multiple-input multiple-output (mMIMO)



Citation: Bui, T.T.; Nguyen, L.D.; Kha, H.H.; Vo, N.-S.; Duong, T.Q. Joint Clustering and Resource Allocation Optimization in Ultra-Dense Networks with Multiple Drones as Small Cells Using Game Theory. *Sensors* **2023**, *23*, 3899. <https://doi.org/10.3390/s23083899>

Academic Editor: Cicek Cavdar

Received: 21 February 2023

Revised: 28 March 2023

Accepted: 10 April 2023

Published: 11 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The global mobile data traffic is witnessing an exponential increase, and the number of global mobile connected devices is forecast to increase rapidly from 8.8 billion in 2018 to 13.1 billion in 2023 [1]. The deployment of ultra-dense networks (UDNs) as a type of cellular network is an extremely promising technology for supporting the huge number of connections, especially in the applications of massive Internet of Things (IoT). In comparison with present network technologies, UDNs help to considerably improve the network capacity, spectral efficiency, as well as coverage with a high quality-of-service (QoS) by reducing the distance between the base stations (BSs) and user equipment (UE) [2,3]. However, the highly dense deployment of small cells in UDNs causes high interference, high power consumption, and high computational complexity, which must be addressed to satisfy the stringent constraints of the beyond fifth-generation networks (B5G) [4].

In UDNs, small cells are smaller than ever, and therefore the distance of neighboring small cells decreases. As a result, the intercell interference is extremely high. In addition,

the interference is especially hard to control because small cells are randomly distributed. There are several studies on interference mitigation in UDNs. In [5], to cope with co-channel interference, the authors used a conflict graph based on machine learning to form a clustering problem and subchannel allocation (SCA) problem for throughput maximization. Two other user-centric clustering methods, using k-means algorithms and jamming strategies, were proposed in [6,7]. These papers show that the clustering-based approach is an efficient tool for mitigating interference in UDNs. However, the clustering methods in [5–7] did not consider the signal to interference plus noise ratio (SINR), which directly influences the data rate. This can lead to a decline in the throughput, which is an important advantage of UDNs compared with conventional networks. In [8], an adaptive clustering method was proposed to maximize the spectral efficiency and the network throughput using two stages: offline and online. Additionally, sum-rate maximization was taken into account as the main objective in a user-centric clustering method, a modified Louvain method, proposed in [9].

Owing to maneuverability, using unmanned aerial vehicles (UAVs) in communications is becoming increasingly popular because the networks can be quickly built and have flexibility. Compared with small terrestrial base stations, drones can easily move to form many different structures of networks to serve users more efficiently. Additionally, the channels between drones and users are usually line-of-sight channels because drones fly at high altitudes where obstacles to the users are less likely. It is also convenient for network operators because small cells can be easily added or removed to increase performance or save cost depending on the number of connections as well as the requirements of users. Thus, in communications, using UAVs as small flying base stations is a promising solution to serve hotspots where there is a massive number of users such as at sports events, concerts, exhibitions, and fairs, or provide mission-critical services, e.g., disaster recovery, traffic congestion, etc. [10,11]. UAVs, as flying base stations, can be widely used in many scenarios such as cognitive radio networks [11], ultra-reliable low-latency communications [12], communications using reconfigurable intelligent surface [13], and caching [14]. Despite the advantages of UAVs compared with terrestrial base stations, they still face many challenges in the attempts at employing UAVs in wireless networks. For example, in terms of deployment, the authors of [15] proposed a deployment method for multiple UAVs with the aim of maximizing both the coverage area and the coverage lifetime, whereas a Q-learning method with the reward being the sum of user data rates was used in [16]. Regarding area throughput and energy management, mixed-integer programming problems were formulated and efficiently solved for establishing the optimal location and action scheduling under the constraints of battery and powered energy in [17,18]. Additionally, the authors of [19] considered improving the spectrum efficiency by optimizing bandwidth allocation, power allocation, and trajectory for one UAV. On the other hand, due to the fast movements and jitters of UAVs, there are quick changes in the channels between the UAVs and ground users [20]. Therefore, real-time computing is crucial in UAV-assisted networks, especially in UDNs with many small cells [11,21].

In UDNs, the complexity of optimal resource allocation problems is very high because massive numbers of small cells and small UEs (SUEs) are randomly and densely deployed in large-scale areas. Game theory (GT), which is a distributed mathematical framework, is efficiently used for decoupling an extremely complex problem into multiple subproblems with lower complexity. Thus, there are several studies that investigated the applications of GT in UDNs. In [22], two noncooperative games (NCGs) were designed for solving the power allocation and user association with constraints on the quality of service (QoS). The proposed combination of backhaul game and access game significantly improved the total rate compared with the hierarchical game. Additionally, in [23], the authors considered small cell networks with nonorthogonal multiple access (NOMA) with the aim of network throughput maximization. The results showed the effectiveness of NOMA and the locally cooperative game in interference mitigation in UDNs. In spite of some improvements in [22,23], the main objective was to maximize the throughput; in UDNs, the energy issue is more critical due to the high power consumption of a massive number of

network elements. Energy efficiency (EE), which is defined as the ratio of total data rate to total power consumption, is a useful metric for modeling the trade-off between throughput and energy. However, EE maximization problems are nonconvex optimization problems and much harder than throughput maximization problems or power minimization problems. In [24,25], two optimization problems (i.e., SCA and power allocation (PA)) were considered with PA problems modeled by the Stackelberg game (SG) in order to maximize the EE. Although there were some considerable improvements in terms of EE, there was a limitation in the applicability of the used model of UDNs, which had only one macro base station (MBS) and one BS in a network. In the previous works, either cooperation or noncooperation between network elements was considered, but these elements have the characteristics of both collaboration and competition. In detail, small-cell base stations (SBSs) prefer to be in joint clusters for cooperatively transmitting signals to UEs. This leads to a decrease in intracluster interference. On the other hand, the SBSs in different clusters tend to conflict regarding power usage. This means that increasing the power of an SBS in a cluster also increases the interference for SUEs in adjacent clusters. Therefore, there is a lack of a game-based paradigm that has the characteristics of both cooperation and competition of network elements in order to concurrently mitigate interference, maximize the network EE in drone-aided UDNs with multiple MBSs, and handle a massive number of randomly distributed drones.

Motivated by the aforementioned discussion, this paper proposes a mathematical model to jointly design the clustering and resource allocation for maximizing the total network EE in UDNs consisting of multiple MBSs and many small-cell drones (SCDs). Cells served by MBSs are macrocells, and cells served by drones are small cells. We propose a coalition game for clustering SCDs to restrict the intercell interference. The SCDs in each cluster cooperatively transmit the signals to UEs. The SCA methods for two tiers are provided to choose the subchannel with the highest channel gain for each UE. To reduce the computational complexity, centralized and distributed PA methods are proposed in terms of maximizing the EE. The main contributions of our paper are summarized as follows:

- Adjacent SCDs tend to join together and cooperatively serve UEs because they highly interfere with each other. Thus, we form this cooperation with a coalition game, with the utility function being the signal-to-interference ratio (SIR).
- To fully eliminate the intracluster interference, we model the SCA for small cells as a binary optimization problem. Then, the Hungarian method is proposed to obtain the solution. Meanwhile, the SCA for macrocells chooses the subchannel with the highest channel gain without any constraint.
- To reduce the complexity of nonconvex PA optimization problems, we propose an iterative centralized PA algorithm to obtain at least a locally optimal solution in terms of maximizing the EE. Inequalities are used for relaxing the complex objective functions of EE maximization into convex functions that are easy to solve with programming tools.
- The distributed PA algorithm based on SG consists of two subcooperative games with MBSs as leaders and SCDs as followers. This algorithm decouples the very complex PA optimization problems into multiple low-complexity and convex ones that are extremely useful for solving optimization problems with a huge number of variables.

Compared with the conference version [26], a macrocell layer consisting of multiple MBSs is added to the networks. As such, the interference environment is more complex due to intercell interference. Additionally, the objective function is to minimize the network EE, which is more practical to use energy more efficiently compared with the objective of maximizing the sum-rate in UDNs but is more complex to solve because of the fractional-type function. Moreover, we propose a centralized approach to solve PA as a benchmark and the distributed one using the Stackelberg game, and multiple simulations with different aspects such as EE, sum rate, and execution time were performed to prove the efficiency of the proposed methods. The rest of this paper is organized as follows: Section 2 presents the system model and optimization problems used in this paper. A small-cell clustering method

based on the coalition game is described in Section 3. Then, the SCA for both macrocells and small cells as well as the beamforming method for MBSs are discussed in Section 4. In addition, Section 5 describes both centralized and distributed optimization methods for solving PA problems. Numerical simulations proving the efficiency of our proposed methods are outlined in Section 6. Section 7 provides the conclusions of this study.

2. System Model and Problem Formulation

2.1. System Model

In this paper, we consider the downlink transmission in a two-tier UDN that consists of multiple macrocells and a massive number of UAVs serving as small-cell base stations with highly dense deployments. MBSs and SCDs share the same N orthogonal subchannels with bandwidth BW per subchannel to serve their UEs. The advantage of using multiple orthogonal subchannels is that multiple UEs can be simultaneously served in these subchannels without interference. One typical macrocell with two subchannel is illustrated in Figure 1. In the macrocell tier, each F MBS is equipped with T array antennas to transmit the signal to M MUEs by using massive multiple-input multiple-output (mMIMO) techniques. In the small-cell tier, to serve U UEs, many B SCDs equipped with one omni-directional antenna each are randomly deployed according to the independent homogeneous Poisson point processes (PPP) Φ with density λ in the coverage of MBSs ($B > U$) [24]. When UEs are located in the coverage of one or some SCDs, they are served by SCDs, i.e., SUEs. On the other hand, if the UEs are not in the coverage of any SCD, then the power of SCDs is not high enough to serve these UEs or causes extremely high interference with surrounding SCDs, especially in UDNs. Therefore, they are served by MBSs, i.e., MUEs. We used the main mathematical notation, which is shown in Table 1, to design the optimization problems. At each MBS, because of the mMIMO, the number of antennas of MBS f is much greater than the number of its authorized MUEs ($T \gg M_f$). Each signal $s_{m,f}^{(n)}$ is multiplied by a beamforming vector $w_{m,f} \in \mathbb{C}^{T \times 1}$, and then the aggregation of processed signals of MUEs that are served by MBS f is transmitted.

Table 1. The description of the main mathematical notation.

Notation	Definition
$\mathcal{F}$	$\{1, 2, \dots, F\}$: The set of MBSs
$\mathcal{B}$	$\{1, 2, \dots, B\}$: The set of SCDs
$\mathcal{C}$	$\{1, 2, \dots, C\}$: The set of clusters of SCDs
$\mathcal{C}_i$	$\{1, 2, \dots, C_i\}$: The set of SCDs in cluster i
$\mathcal{U}$	$\{1, 2, \dots, U\}$: The set of SUEs
$\mathcal{M}$	$\{1, 2, \dots, M\}$: The set of MUEs
$\mathcal{U}_{C_i}$	$\{1, 2, \dots, U_{C_i}\}$: The set of SUEs served by cluster C_i
$\mathcal{U}_b$	$\{1, 2, \dots, U_b\}$: The set of SUEs served by SCD b
$\mathcal{M}_f$	$\{1, 2, \dots, M_f\}$: The set of MUEs served by MBS f
$s_{m,f}^{(n)}$	$\in \mathbb{C}$: The instant desired symbol of MUE m served by MBS f in subchannel n
$p_{m,f}^{(n)}$	$\mathbb{E}\{s_{m,f}^{(n)} s_{m,f}^{(n)H}\}$: The transmit power of MBS f to MUE m in subchannel n
$s_{u,b}^{(n)}$	$\in \mathbb{C}$: The instant desired symbol of SUE m served by SCD b in subchannel n

Table 1. Cont.

Notation	Definition
$p_{u,b}^{(n)}$	$\mathbb{E}\{s_{u,b}^{(n)} s_{u,b}^{(n)H}\}$: The transmit power of SCD b to SUE u in subchannel n
$h_{m,f}^{(n)}$	$\in \mathbb{C}^{1 \times T}$: The channel response from MBS f to MUE m in subchannel n
$h_{m,b}^{(n)}$	$\in \mathbb{C}$: The channel response from interference source SCD b to MUE m in subchannel n
$h_{u,b}^{(n)}$	$\in \mathbb{C}$: The channel response from SCD b to SUE u in subchannel n
$h_{u,f}^{(n)}$	$\in \mathbb{C}^{1 \times T}$: The channel response from interference source MBS f to SUE u in subchannel n
$\mathbf{w}_{m,f}$	$\in \mathbb{C}^{T \times 1}$: Beamforming vector at MBS f to serve MUE m
$a_{m,f}^{(n)}, a_{u,C_i}^{(n)}$	subchannel allocation indicator
$y_{m,f}^{(n)}$	Received signal at MUE m served by MBS f in subchannel n
$y_{u,C_i}^{(n)}$	Received signal of SUE u served by cluster C_i in subchannel n
$I_{u,\mathcal{F}}^{(n)}$	$\sum_{f \in \mathcal{F}} \sum_{m \in \mathcal{M}_f} a_{m,f}^{(n)} h_{u,f}^{(n)} \mathbf{w}_{m,f} ^2 p_{m,f}^{(n)}$: The interference power from MBSs
$I_{u,\mathcal{C} \setminus C_i}^{(n)}$	$\sum_{C_j \in \mathcal{C} \setminus C_i} \sum_{b' \in C_j} \sum_{u' \in \mathcal{U}_{C_j}} a_{u',C_j}^{(n)} h_{u,b'}^{(n)} ^2 p_{u',b'}^{(n)}$: The interference power from other clusters except cluster C_i
$I_{\text{cross-talk}}^{(n)}$	$\sum_{m' \in \mathcal{M}_f \setminus m} a_{m',f}^{(n)} h_{m,f}^{(n)} \mathbf{w}_{m',f} ^2 p_{m',f}^{(n)}$: The cross-talk interference power from MBS f
$I_{m,\mathcal{F} \setminus f}^{(n)}$	$\sum_{f' \in \mathcal{F} \setminus f} \sum_{m' \in \mathcal{M}_{f'}} a_{m',f'}^{(n)} h_{m,f'}^{(n)} \mathbf{w}_{m',f'} ^2 p_{m',f'}^{(n)}$: The interference power from MBSs except MBS f
$I_{m,\mathcal{C}}^{(n)}$	$\sum_{C_i \in \mathcal{C}} \sum_{b \in C_i} \sum_{u \in \mathcal{U}_{C_i}} a_{u,C_i}^{(n)} h_{m,b}^{(n)} ^2 p_{u,b}^{(n)}$: The interference power from the clusters of SCDs
$\mathcal{CN}(0, \sigma^2)$	A complex Gaussian random variable with zero mean and variance σ^2

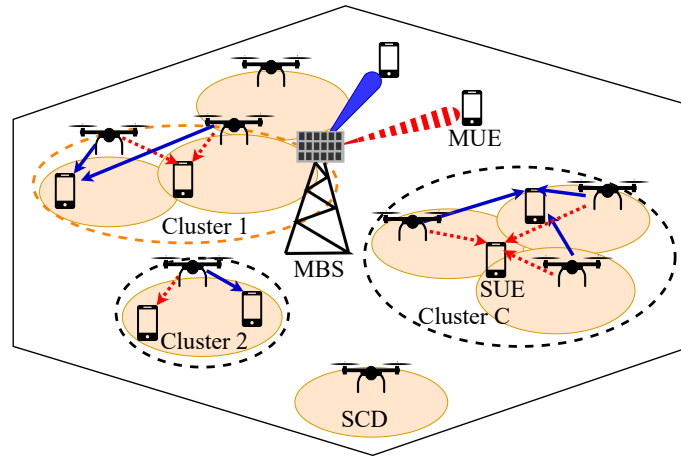


Figure 1. A typical hexagonal macrocell in UDNs. Each small cell has an SCD as a flying base station with bounded circle coverage. Dotted red lines and beams denote the signal from BSs to UEs in a subchannel, while the others denote the signal in another one.

The aggregation $\mathbf{x}_f^{(n)}$ at MBS f in subchannel n is the sum of M_f processed signals:

$$\mathbf{x}_f^{(n)} = \sum_{m \in \mathcal{M}_f} a_{m,f}^{(n)} \mathbf{w}_{m,f} s_{m,f}^{(n)} \quad (1)$$

where the binary indicator of SCA is defined as

$$a_{m,f}^{(n)} = \begin{cases} 1, & \text{if MUE } m \text{ is assigned MBS } f \text{ in subchannel } n, \\ 0, & \text{otherwise.} \end{cases}$$

The received signal $y_{m,f}^{(n)}$ at MUE m served by MBS f in subchannel n is expressed by

$$y_{m,f}^{(n)} = \underbrace{h_{m,f}^{(n)} w_{m,f} s_{m,f}^{(n)}}_{\text{desired signal}} + \underbrace{\sum_{m' \in \mathcal{M}_f \setminus m} a_{m',f}^{(n)} h_{m',f}^{(n)} w_{m',f} s_{m',f}^{(n)}}_{\text{cross-talk interference}} + \underbrace{\sum_{f' \in \mathcal{F} \setminus f} h_{m,f'}^{(n)} x_{f'}^{(n)}}_{\text{interference from other MBSs}} + \underbrace{\sum_{C_i \in \mathcal{C}} \sum_{b \in C_i} \sum_{u \in \mathcal{U}_{C_i}} a_{u,C_i}^{(n)} h_{u,b}^{(n)} s_{u,b}^{(n)}}_{\text{interference from clusters}} + n_m, \quad (2)$$

where the aggregation of signals at MBS f' is $x_{f'}^{(n)} = \sum_{m' \in \mathcal{M}_{f'}} a_{m',f'}^{(n)} w_{m',f'} s_{m',f'}^{(n)}$, and $n_m \sim \mathcal{CN}(0, \sigma^2)$ is additive white Gaussian noise (AWGN) at MUE m .

The received signal $y_{u,C_i}^{(n)}$ of SUE u served by cluster C_i in subchannel n is given by

$$y_{u,C_i}^{(n)} = \underbrace{\sum_{b \in C_i} h_{u,b}^{(n)} s_{u,b}^{(n)}}_{\text{desired signal}} + \underbrace{\sum_{f \in \mathcal{F}} h_{u,f}^{(n)} x_f^{(n)}}_{\text{interference from MBSs}} + \underbrace{\sum_{C_j \in \mathcal{C} \setminus C_i} \sum_{b' \in C_j} \sum_{u' \in \mathcal{U}_{C_j}} a_{u',C_j}^{(n)} h_{u,b'}^{(n)} s_{u',b'}^{(n)}}_{\text{interference from other clusters}} + n_u, \quad (3)$$

where $n_u \sim \mathcal{CN}(0, \sigma^2)$ is the AWGN of SUE u , $h_{u,b}^{(n)} = g_{u,b}^{(n)} \sqrt{\beta_{u,b}^{(n)}}$ is the channel response, $g_{u,b}^{(n)}$ is a complex Gaussian random variable with zero mean and unit variance that represents the small-scale fading of the channel from SCD b to SUE u , and $\beta_{u,b}^{(n)}$ is the large-scale coefficient of this channel. Therefore, the SINRs at SUE u and MUE m are, respectively, written as

$$\gamma_u = \frac{\sum_{b \in C_i} |h_{u,b}^{(n)}|^2 p_{u,b}^{(n)}}{I_{u,\mathcal{F}}^{(n)} + I_{u,\mathcal{C} \setminus C_i}^{(n)} + \sigma_u^2}, \quad (4)$$

$$\gamma_m = \frac{|h_{m,f}^{(n)} w_{m,f}|^2 p_{m,f}^{(n)}}{I_{\text{cross-talk}}^{(n)} + I_{m,\mathcal{F} \setminus f}^{(n)} + I_{m,\mathcal{C}}^{(n)} + \sigma_m^2}. \quad (5)$$

2.2. Problem Formulation

As the deployment of small cells in UDNs becomes extremely dense, the problem of effectively using resources becomes more important than ever. In this section, we formulate the optimization problems for the resource allocation of both spectrum and power according to the processing operations in Figure 2. The overall architectures of the proposed optimization solutions are shown in Figure 2 with the names of the 6 problems in blocks. The detailed description and algorithms are presented in Sections 3–5. The objective functions at different stages are different to obtain an optimal solution for multiple proposed objectives, such as interference mitigation in clustering, a guarantee of high-gain channels in subchannel allocation, and efficient energy usage in power allocation.

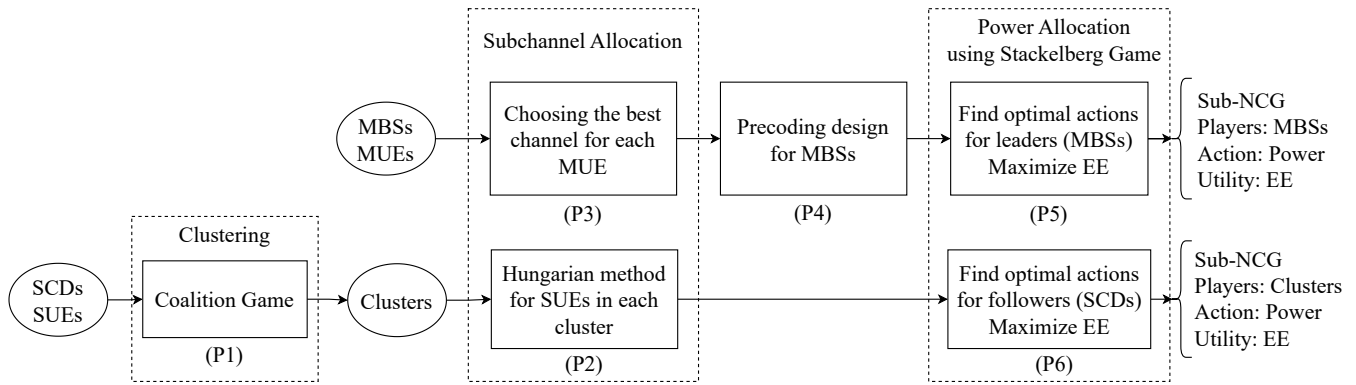


Figure 2. Block processing operations of proposed optimization solutions.

2.2.1. Subchannel Allocation (SCA)

Due to the high efficiency of spectral usage, co-channel assignment is usually used in practical systems [27]. However, this causes high interference between the two tiers in UDNs and is hard to control, especially with the random deployment of small cells. Thus, the optimization problems with the objective to choose the highest channel gain guarantee that the BSs serve their UEs in good subchannels.

SCA for macrocells: We consider MBS f serving M_f MUEs. By using mMIMO technology at each MBS, the number of MUEs in each subchannel causes a minor interference effect. The target is to find a subchannel with the highest gain for each MUE. The optimization problem can be expressed as

$$\max_{a_{m,f}^{(n)}} \sum_{m=1}^{M_f} \sum_{n=1}^N VM_{m,n} a_{m,f}^{(n)} \quad (6a)$$

$$\text{s.t.} \quad \sum_{n=1}^N a_{m,f}^{(n)} = 1, m = 1, \dots, M_f, \quad (6b)$$

$$a_{m,f}^{(n)} \in \{0, 1\}, n = 1, \dots, N, \forall m \in \mathcal{M}_f, \quad (6c)$$

where $VM_{m,n} = \text{trace} \left\{ \left\| \mathbf{h}_{m,f}^{(n)} \right\|^2 \right\}$. (6b) indicates that each MUE $m \in \mathcal{M}_f$ authorizes only one subchannel. (6c) represents the constraint of the binary values of the SCA. If $a_{m,f}^{(n)} = 1$, then MUE m is served by MBS f through subchannel n , and vice versa if $a_{m,f}^{(n)} = 0$. Optimization problem (6) is denoted by (P3) in Figure 2. Problem (6) is integer linear programming because (6a) and (6b) are affine functions, and (6c) is the constraint of two discrete values of variables.

SCA for small cells: In Figure 2, after clustering the small cells, the SCDs in each cluster cooperatively transmit the same signal to their SUEs. Therefore, the power of the desired signal at each SUE is improved compared with that using only one serving SCD. Considering cluster $\mathcal{C}_i$, to find the appropriate subchannels for the SUEs in set $\mathcal{U}_{\mathcal{C}_i}$, an optimization problem is formulated as

$$\max_{a_{u,\mathcal{C}_i}^{(n)}} \sum_{u=1}^{\mathcal{U}_{\mathcal{C}_i}} \sum_{n=1}^N VC_{u,n} a_{u,\mathcal{C}_i}^{(n)} \quad (7a)$$

$$\text{s.t.} \quad \sum_{u=1}^{\mathcal{U}_{\mathcal{C}_i}} a_{u,\mathcal{C}_i}^{(n)} = 1, n = 1, \dots, N, \quad (7b)$$

$$\sum_{n=1}^N a_{u,\mathcal{C}_i}^{(n)} = 1, \forall u \in \mathcal{U}_{\mathcal{C}_i}, \quad (7c)$$

$$a_{u,\mathcal{C}_i}^{(n)} \in \{0, 1\}, n = 1, \dots, N, \forall u \in \mathcal{U}_{\mathcal{C}_i}, \quad (7d)$$

where $VC_{u,n} = \sum_{b \in \mathcal{C}_i} |h_{u,b}^{(n)}|^2$ is the sum of channel gains from SCDs in cluster $\mathcal{C}_i$ to SUE u . (7b), (7c) indicates that each subchannel n is authorized by at most one SUE $u \in \mathcal{U}_{\mathcal{C}_i}$ and vice versa. (7d) is the constraint of the binary values of SCA. If $a_{u,\mathcal{C}_i}^{(n)} = 1$, then SUE u is served by the SCDs in cluster $\mathcal{C}_i$ through subchannel n , and vice versa if $a_{u,\mathcal{C}_i}^{(n)} = 0$. Optimization problem (7) is denoted by (P2) in Figure 2. Problem (7) is integer linear programming because (7a)–(7c) are affine functions, and (7d) is the constraint of two discrete values of variables. The solving methods for (P2) and (P3) are proposed in Section 4.1.

2.2.2. Power Allocation (PA)

In UDNs, the distance between the SCDs and SUEs as well as the distance between small cells are very small: from a few dozen to a few hundred meters. Therefore, increasing the power at an SCD to improve the signal to its UEs causes extremely high interference to other small cells compared with that of conventional networks. In addition, different tiers have different objectives that need to be optimized. For that reason, after SCA, we formulate two optimization problems for PA in macrocells and small cells (Figure 2).

PA for MBSs (EE maximization): Each MBS, equipped with massive antennas, is connected directly to a core network. MBSs, using high power with wide coverage areas, guarantee the serving of all UEs that are not connected to any small cell. Therefore, an important problem for MBSs is maximizing the data rate of the MUEs to satisfy the standards of the new generations of future networks (B5G) while efficiently using energy. EE, which is the ratio of data rate to power consumption, is widely used for evaluating the efficiency of energy usage. The optimization problem of PA for MBSs is as follows

$$\max_{p_{m,f}^{(n)}} \frac{\sum_{m=1}^M BW \log_2(1 + \gamma_m)}{\sum_{f \in \mathcal{F}} P_f^{\text{total}}} \quad (8a)$$

$$\text{s.t.} \quad \sum_{n=1}^N \sum_{m \in \mathcal{M}_f} a_{m,f}^{(n)} p_{m,f}^{(n)} \leq P_{\text{MBS}}^{\max}, \forall f \in \mathcal{F}, \quad (8b)$$

$$p_{m,f}^{(n)} \geq 0, \forall f \in \mathcal{F}, \forall m \in \mathcal{M}, n = 1, \dots, N, \quad (8c)$$

$$BW \log_2(1 + \gamma_m) \geq T p_{m,\min}, \forall m \in \mathcal{M}, \quad (8d)$$

where the variable $p_{m,f}^{(n)}$ represents the transmit power of MBS f to MUE m in subchannel n . $P_f^{\text{total}} = P_f^{\text{circuit}} + \sum_{n=1}^N \sum_{m \in \mathcal{M}_f} a_{m,f}^{(n)} p_{m,f}^{(n)}$ is the sum of power for the hardware circuits and power for transmitting signal in all subchannels of MBS f , $P_{\text{MBS}}^{\max}$ is the maximum power of each MBS, and $T p_{m,\min}$ is the minimum throughput of each MUE. (8b) and (8c) are the constraints of the transmit power of MBSs, which range from 0 to $P_{\text{MBS}}^{\max}$. (8d) guarantees the QoS of all UEs.

PA for SCDs (EE maximization): Due to the short distance between the SCDs and SUEs, the channel links experience the line of sight (LoS). However, with a large number of small cells, the power consumption rises considerably. Thus, the target of SCDs is effectively using limited energy. The optimization problem for maximizing the EE of small cells is formulated as

$$\max_{p_{u,b}^{(n)}} \frac{\sum_{u=1}^U BW \log_2(1 + \gamma_u)}{\sum_{\mathcal{C}_i \in \mathcal{C}} \sum_{b \in \mathcal{C}_i} P_b^{\text{total}}} \quad (9a)$$

$$\text{s.t.} \quad \sum_{n=1}^N \sum_{u \in \mathcal{U}_{\mathcal{C}_i}} a_{u,\mathcal{C}_i}^{(n)} p_{u,b}^{(n)} \leq P_{\text{SCD}}^{\max}, \forall b \in \mathcal{B}, \quad (9b)$$

$$p_{u,b}^{(n)} \geq 0, \forall b \in \mathcal{B}, \forall u \in \mathcal{U}, n = 1, \dots, N, \quad (9c)$$

$$BW \log_2(1 + \gamma_u) \geq T p_{u,\min}, \forall u \in \mathcal{U}, \quad (9d)$$

where $p_{u,b}^{(n)}$ denotes the transmit power of SCD b to SUE u in subchannel n . $P_b^{\text{total}} = P_b^{\text{circuit}} + \sum_{n=1}^N \sum_{u \in \mathcal{U}_{C_i}} a_{u,C_i}^{(n)} p_{u,b}^{(n)}$ is the sum of power for hardware circuits and power for transmitting signal in all subchannels of SCD $b \in \mathcal{C}_i$, and $P_{\text{SCD}}^{\text{max}}$ is the maximum transmit power of each SCD, TP_{min} is the minimum throughput of each MUE. (9b) and (9c) represent the range of transmit power. To guarantee the QoS, (9d) is used for all SUEs.

3. SCDs Clustering by Coalition Game (P1)

In this section, we introduce a coalition game for clustering small cells. In UDNs, an effective clustering method is essential because of three main reasons, as follows:

- Mitigating interference: Intercell interference is a big challenge in UDNs due to the ultra-dense density of small cells, especially for SUEs in the overlapping areas of small cells. Formulating clusters that combine small cells seriously interfering with each other is an appropriate solution. The SCDs in the same cluster cooperatively transmit the signal to UEs so as there is no intracluster interference.
- Enhancing the quality of the signal: In UDNs, the SUEs in the overlapping areas of two or more small cells are usually far from their primal SCDs, while they receive high interference from the others. In many cases, using some of these SCDs along with primary SCDs to serve UEs helps improve the power of the desired signal. Furthermore, multiple SCDs transmitting the same signal to a UE also improves the diversity gain.
- Decreasing handover processes: The smaller the size of small cells, the larger the number of handover processes. Moreover, when UEs move at high speed, the central controller has to compute the complex handover processes with the massive number of UEs in a short time. Therefore, if small cells join together to form temporarily bigger cells, the number of handover processes significantly declines.

Therefore, adjacent small cells have an incentive to join clusters to obtain a higher value for the objective instead of independent transmission. The coalition game, as a type of cooperative game where players tend to cooperate to achieve higher total utility, is appropriate to model this situation. The standard form of a coalition game is $CG = \langle PL, v \rangle$, where $PL = \mathcal{B}_{\text{active}}$ is the set of players (i.e., SCDs, which serve at least one SUEs), and $v(\mathcal{C}) \subseteq \mathbb{R}^C$ is the set of nontransferable utility functions of clusters [28]. The utility function of cluster $\mathcal{C}_i$ is defined as

$$v(\mathcal{C}_i) = \sum_{u \in \mathcal{U}_{C_i}} SIR_u, \quad (10)$$

where SIR_u is the SIR of the links from cluster $\mathcal{C}_i$ to SUE u and is expressed as

$$SIR_u = \frac{\sum_{b' \in \mathcal{C}_i} |\bar{h}_{u,b'}|^2}{\sum_{C_j \in \mathcal{C} \setminus \mathcal{C}_i} \sum_{b \in \mathcal{C}_j} |\bar{h}_{u,b}|^2}, \quad (11)$$

where $\bar{h}_{u,b} = \frac{1}{N} \sum_{n=1}^N h_{u,b}^{(n)}$ is the average of channel gain from SCD b to SUE u . The solution of this coalition game is the set of coalitions of players or clusters of SCDs in UDNs [28]. In these clusters, there is no SCD that has an incentive to change clusters for achieving higher utility. This is the state where all clusters have reached equilibrium. The two operations that change the SCDs between clusters are split-merge and swap [29]. If SCD $b \in \mathcal{C}_j$ leaves $\mathcal{C}_j$ to join $\mathcal{C}_i$, then the operation is denoted as $\{\mathcal{C}_i, \mathcal{C}_j\} \rightarrow \{\mathcal{C}_i \cup \{b\}, \mathcal{C}_j \setminus \{b\}\}$. In addition, if SCD $b \in \mathcal{C}_j$ swaps with $b' \in \mathcal{C}_i$, then the operation is described as $\{\mathcal{C}_i, \mathcal{C}_j\} \rightarrow \{\mathcal{C}_i \cup \{b\} \setminus \{b'\}, \mathcal{C}_j \cup \{b'\} \setminus \{b\}\}$. SCD b tends to leave $\mathcal{C}_j$ to join $\mathcal{C}_i$ or prefer $\mathcal{C}_i$ to $\mathcal{C}_j$ if $v(\mathcal{C}_i \cup \{b\}) + v(\mathcal{C}_j \setminus \{b\}) > v(\mathcal{C}_i) + v(\mathcal{C}_j)$.

To obtain the solution to this coalition game, we use Algorithm 1. In this algorithm, we use the variable d_{max} for determining the SUEs that are less than d_{max} m away from SCD b . Then, the set of neighbor clusters $\mathcal{C}_{\text{near}}$ that consists of these SUEs is determined.

Therefore, different from [30], we only consider some close clusters with SCD b because the interference from b has a minor impact on far SUEs at very high frequency. In addition, variables g and g_n are used for choosing the best option for SCD b to move to a new cluster. If the number of SCDs in each cluster increases, then the number of pairs of SCD and SUE with a distance larger than $d_{\max}$ increases. Thus, we use $N_{\max}$ for limiting the number of SCDs in each cluster.

Algorithm 1 Clustering SCDs using the coalition game.

```

1: Input: Channel matrix  $\mathbf{H}$ , the set of players  $\mathbf{PL} = \mathcal{B}_{\text{active}}$ , locations of SCDs and SUEs,
   the maximum number of SCDs in a cluster  $N_{\max}$ , the number of subchannels  $N$ ,  $d_{\max}$ 
2: Randomly initialize a partition  $\mathcal{C}^{(0)}$ ,  $l = 0$ 
3: while  $\mathcal{C}^{(l)} \neq \mathcal{C}^{(old)}$ 
4:    $\mathcal{C}^{(old)} \leftarrow \mathcal{C}^{(l)}$ 
5:   for  $b \in \mathcal{B}_{\text{active}}$ 
6:     Assume  $b \in \mathcal{C}_j$ ,  $\mathcal{C}_{\text{near}}$  is the set of clusters that have some SUEs covered with radius
        $d_{\max}$  and center  $b$ , gap  $g = 0$ 
7:     for  $\mathcal{C}_i \in \mathcal{C}_{\text{near}}$ 
8:       if  $|\mathcal{C}_i| = N_{\max}$  and  $U_{\mathcal{C}_i} + U_b - U_{b'} \leq N$  and  $U_{\mathcal{C}_j} + U_{b'} - U_b \leq N$ 
9:          $\mathcal{C}^{(tmp)} \leftarrow$  swap SCD  $b$  and SCD  $b' \in \mathcal{C}_i$ 
10:         $g_n = v(\mathcal{C}_i \cup \{b\} \setminus \{b'\}) + v(\mathcal{C}_j \cup \{b'\} \setminus \{b\}) - v(\mathcal{C}_i) - v(\mathcal{C}_j)$ 
11:       else if  $U_{\mathcal{C}_i} + U_b \leq N$ 
12:          $\mathcal{C}^{(tmp)} \leftarrow$  SCD  $b$  joins  $\mathcal{C}_i$ 
13:         $g_n = v(\mathcal{C}_i \cup \{b\}) + v(\mathcal{C}_j \setminus \{b\}) - v(\mathcal{C}_i) - v(\mathcal{C}_j)$ 
14:       if  $g_n > g$ 
15:          $\mathcal{C}^{(ok)} \leftarrow \mathcal{C}^{(tmp)}$ ,  $g = g_n$ 
16:        $\mathcal{C}^{(l)} \leftarrow \mathcal{C}^{(ok)}$ 
17:      $l \leftarrow l + 1$ 
18: Output: The set of clusters  $\mathcal{C}$ 

```

4. Subchannel Allocation (P2, P3) and Beamforming Design for MBSs (P4)

In this section, we propose methods to solve the SCA problems (6) and (7). Satisfactory subchannels are allocated to UEs with high channel gain while guaranteeing the constraints are met. In addition, the beamforming design for MBSs is also described.

4.1. Subchannel Allocation

(P2) For clusters, constraints (7b)–(7d) are the same as the constraints of an assignment problem where each task is assigned to only one worker to minimize the cost. One of the effective methods to solve assignment problems is the Hungarian method [31]. Compared with other methods, the Hungarian algorithm can solve large-scale assignment problems in polynomial time. Additionally, it is a deterministic and global algorithm that guarantees the optimal solution is the best assignment. Regarding flexibility, the Hungarian method can be applied to solve assignment problems with equal or unequal numbers of resources and tasks, especially in the cases of variable numbers of network elements in clusters in our proposed SCA problem. Therefore, we use the Hungarian algorithm for making decisions in choosing the optimal subchannels for SUEs. Matrix $\mathbf{VC}_{\max\text{prob}}$, which consists of the value of $\mathbf{VC}_{u,n}$ with all pairs of SUE $u \in \mathcal{C}_i$ and subchannel n , is given in Table 2.

Table 2. The matrix of values of $VC_{u,n}$.

SUE\Channel	1	2	...	N
1	$VC_{1,1}$	$VC_{1,2}$	...	$VC_{1,N}$
2	$VC_{2,1}$	$VC_{2,2}$	...	$VC_{2,N}$
...	...	...	...	...
U_{C_i}	$VC_{U_{C_i},1}$	$VC_{U_{C_i},2}$	...	$VC_{U_{C_i},N}$

To obtain the solution to the optimization problem (7), we use the Hungarian method in Algorithm 2 [30]. Because of the limitation of the number of UEs in each cluster by N , the complexity of Algorithm 2 is $O(N^4)$ [31].

Algorithm 2 The Hungarian method for SCA in each cluster.

- 1: **Input:** VC matrix $VC_{maxprob}$.
 - 2: Convert to minimization problem by subtracting the maximum of the matrix from all elements $VC_{minprob} = \{VC_{i,k} = |VC_{i,k} - \max(VC_{maxprob})| \mid i = 1, \dots, U_{C_i} \text{ and } k = 1, \dots, N\}$.
 - 3: Extend $VC_{minprob}$ to a square matrix with the size of $N \times N$
 - 4: **Obtain modified matrix:**
 - 5: Subtract the smallest entry from all entries in each row of $VC_{minprob}$.
 - 6: Subtract the smallest entry from all entries in each column of $VC_{minprob}$.
 - 7: **Repeat**
 - 8: Cover all zeros in the matrix with minimum number of vertical/horizontal lines nl_{min} .
 - 9: **if** $nl_{min} < N$ **then**
 - 10: Subtract the lowest number from all elements that are not covered by any line.
 - 11: Add this lowest number to elements that are crossed by any two lines.
 - 12: **Until** $nl_{min} = N$.
 - 13: Obtain the optimal association from zeros in the processed matrix.
 - 14: **Output:** The association matrix $X = \{x_{i,k} \mid i = 1, \dots, U_{C_i} \text{ and } k = 1, \dots, N\}$.
-

(P3) For macrocells, constraint (6b) requires one MUE to be allocated only one subchannel. Moreover, the number of MUEs in a subchannel is not important due to the mMIMO MBSs. Therefore, the solution to the optimization problem (6) can be easily obtained by choosing the maximum value of $VM_{m,n}, n = 1, \dots, N$ for each MUE m .

4.2. Beamforming Design

According to the model in Figure 2, after SCA, we need to design the beamforming vectors for MBSs. Assuming that the CSI is known at the MBSs, we introduce two types of appropriate beamforming vectors for two cases.

- If there are multiple MUEs that are served by MBS f in subchannel n , then the beamforming vector is one vector in the null space of the channel matrix from MBS f to all of its MUEs except MUE m , and is expressed as

$$w_{m,f} = \text{Null} \left(H_{\mathcal{M}_f \setminus m, f}^{(n)} \right).$$

Therefore, we can neglect the cross-talk interference in (2).

- If only one MUE m is served by MBS f in subchannel n , then according to the zero-forcing beamforming, the beamforming vector is as follows

$$w_{m,f} = \left(h_{m,f}^{(n)} \right)^\dagger / \left\| \left(h_{m,f}^{(n)} \right)^\dagger \right\|.$$

In both cases, the magnitude of the beamforming vector is normalized to 1 ($\|w_{m,f}\| = 1$). Thus, the PA for MBSs and SCDs only depends on $p_{m,f}^{(n)}$ and $p_{u,b}^{(n)}$.

5. Power Allocation for Macrocells and Small Cells (P5, P6)

Power allocation (PA) is a promising approach for interference mitigation in UDNs. In this section, we propose both a distributed method and a centralized method to obtain the solutions to the optimization problems (8) and (9). It is clear that the powers of MBSs and SCDs are variables in both optimization problems (8) and (9). This causes these optimization problems to be extremely complex and hard to solve. In addition, the channels from MBSs to MUEs are usually more strongly affected by environmental factors than the channel between SCDs and SUEs, while guaranteeing the QoS. Therefore, the optimization problem of MBSs has higher priority than that of SCDs. We assume that the interference from small-cell tiers to each MUE m is a constant at first as follows:

$$I_{m,\mathcal{C}}^{(n)} = \sum_{\mathcal{C}_i \in \mathcal{C}} \sum_{b \in \mathcal{C}_i} \sum_{u \in \mathcal{U}_{\mathcal{C}_i}} a_{u,\mathcal{C}_i}^{(n)} |h_{m,b}^{(n)}|^2 P_{SCD}^{\max} / N. \quad (12)$$

Each active SCD uses the maximum power of $P_{SCD}^{\max}$, which is divided equally to the sub-channels, to serve the SUEs. Thus, the interference from the small-cell tier to MUEs is extremely high when first solving the PA optimization problem of MBSs. This helps the QoS constraints to remain satisfied after obtaining the solution to the PA optimization problem of SCDs with very low interference.

5.1. Centralized Power Allocation

In this subsection, we propose practical methods to solve centralized PA problems for macrocells and small cells. In these methods, we combine three techniques: changing variables, approximating objective functions, and the Dinkelbach algorithm. To this end, the very complex maximizing EE optimization problems especially in UDNs are converted into much easier convex problems in iterative algorithms. This is a promising approach to achieving real-time computing in multitier massive-cell networks such as UDNs.

The EE maximization PA problem for macrocells is expressed as follows

$$\max_{p_{m,f}^{(n)}} \frac{\sum_{m=1}^M \log_2 \left(1 + \frac{|h_{m,f}^{(n)} w_{m,f}|^2 p_{m,f}^{(n)}}{I_{\text{cross-talk}}^{(n)} + I_{m,\mathcal{F}\setminus f}^{(n)} + I_{m,\mathcal{C}}^{(n)} + \sigma_m^2} \right)}{\sum_{f \in \mathcal{F}} P_f^{\text{total}}} = \frac{\varphi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}})}{\pi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}})} \quad (13a)$$

$$\text{s.t.} \quad \sum_{m=1}^N \sum_{m \in \mathcal{M}_f} a_{m,f}^{(n)} p_{m,f}^{(n)} \leq P_{\text{MBS}}^{\max}, \forall f \in \mathcal{F}, \quad (13b)$$

$$p_{m,f}^{(n)} \geq 0, \forall f \in \mathcal{F}, \forall m \in \mathcal{M}, n = 1, \dots, N, \quad (13c)$$

$$\gamma_m \geq \gamma_{m \min}, \forall m \in \mathcal{M}, \quad (13d)$$

where $\gamma_{m \min} \triangleq 2^{T p_{m \min} / BW} - 1$ is the minimum SINR to guarantee the QoS of MUEs. By using the beamforming vectors in Section 4.2, the value of cross-talk interference $I_{\text{cross-talk}}^{(n)}$ in the macrocells equals 0. $I_{m,\mathcal{C}}^{(n)}$ is fixed by using estimation (12). We can easily convert constraint (13d) into a linear inequality because each γ_m in (13d) is an affine-fraction. Therefore, all constraints in the optimization problem (13) are linear. Let $I_m(\mathbf{P}_{\mathcal{F}}) \triangleq I_{\text{cross-talk}}^{(n)} + I_{m,\mathcal{F}\setminus f}^{(n)} + I_{m,\mathcal{C}}^{(n)}$. Definitely, in (13a), a given UE m is served by MBS f in sub-channel n , so $p_{m,f}^{(n)} > 0$ with constraint (13d). For $x_m = |h_{m,f}^{(n)} w_{m,f}|^2 p_{m,f}^{(n)}$, $y_m = I_m(\mathbf{P}_{\mathcal{F}}) + \sigma_m^2$, and at the κ -th iteration $\bar{x}_m = |h_{m,f}^{(n)} w_{m,f}|^2 p_{m,f}^{(n)(\kappa)}$, $\bar{y}_m = I_m(\mathbf{P}_{\mathcal{F}}^{(\kappa)}) + \sigma_m^2$, where $\mathbf{P}_{\mathcal{F}}^{(\kappa)}$ is a feasible point of optimization problem (13). Following the method in [32] and using inequality (A1) in Appendix A.1, we have $\varphi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}) \geq \tilde{\varphi}_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}, \mathbf{P}_{\mathcal{F}}^{(\kappa)})$, where $\tilde{\varphi}_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}, \mathbf{P}_{\mathcal{F}}^{(\kappa)}) \triangleq$

$$\sum_{m=1}^M \left(\tilde{a}_m^{(\kappa)} - \tilde{b}_m^{(\kappa)} / x_m - \tilde{c}_m^{(\kappa)} y_m \right) \text{ and } \tilde{a}_m^{(\kappa)} = \log_2 \left(1 + \frac{\bar{x}_m}{\bar{y}_m} \right) + \frac{2\bar{x}_m}{\ln(2)(\bar{x}_m + \bar{y}_m)}, \tilde{b}_m^{(\kappa)} = \frac{\bar{x}_m^2}{\ln(2)(\bar{x}_m + \bar{y}_m)}, \tilde{c}_m^{(\kappa)} = \frac{\bar{x}_m}{\ln(2)\bar{y}_m(\bar{x}_m + \bar{y}_m)}.$$

Let $\lambda_{\mathcal{M}}^{(\kappa)} \triangleq \varphi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa)}) / \pi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa)})$. According to the Dinkelbach algorithm, the suboptimization problem to find $\mathbf{P}_{\mathcal{F}}^{(\kappa+1)}$ for iteration $(\kappa + 1)$ is as follows

$$\max_{\substack{(n) \\ p_{m,f}}} \tilde{\varphi}_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}, \mathbf{P}_{\mathcal{F}}^{(\kappa)}) - \lambda_{\mathcal{M}}^{(\kappa)} \pi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}) \quad \text{s.t.} \quad (13b), (13c), (13d). \quad (14)$$

Because $\tilde{a}_m^{(\kappa)}, \tilde{b}_m^{(\kappa)}, \tilde{c}_m^{(\kappa)} > 0$, the objective function in (14), which is the sum of pairs of convex functions, i.e., *constant* $\times$ *variable* and $-$ *constant*/*variable* (*constant* > 0), is convex. Therefore, problem (14) is convex due to the convexity of the objective function and the affine functions at the constraints. The proof of the convergence is provided in Appendix B.1. Based on [33], the computational complexity of each iteration for solving (14) is $\mathcal{O}\left((MFN)^2(F + MFN + M)^{2.5} + (F + MFN + M)^{3.5}\right)$ with the number of decision variables MFN and the number of constraints $F + MFN + M$.

It is given that the UE u is served by cluster $\mathcal{C}_i$, which combines SCDs b . The EE maximization PA problem for small cells is given by

$$\max_{p_{u,b}^{(n)}} \frac{\sum_{u=1}^U \log_2 \left(1 + \frac{\sum_{b \in \mathcal{C}_i} |h_{u,b}^{(n)}|^2 p_{u,b}^{(n)}}{I_{u,\mathcal{F}}^{(n)} + I_{u,\mathcal{C} \setminus \mathcal{C}_i}^{(n)} + \sigma_u^2} \right)}{\sum_{i \in \mathcal{C}} \sum_{b \in \mathcal{C}_i} p_b^{\text{total}}} = \frac{\varphi_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}})}{\pi_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}})} \quad (15a)$$

$$\text{s.t.} \quad \sum_{n=1}^N \sum_{u \in \mathcal{U}_{\mathcal{C}_i}} a_{u,\mathcal{C}_i}^{(n)} p_{u,b}^{(n)} \leq p_{\text{SCD}}^{\max}, \forall b \in \mathcal{B}, \quad (15b)$$

$$p_{u,b}^{(n)} \geq 0, \forall b \in \mathcal{B}, \forall u \in \mathcal{U}, n = 1, \dots, N, \quad (15c)$$

$$\gamma_u \geq \gamma_{s \min}, \forall u \in \mathcal{U}, \quad (15d)$$

where $\gamma_{s \min} \triangleq 2^{T p_{s \min} / BW} - 1$ is the minimum SINR that guarantees the QoS of the SUEs. Using the same technique as for the PA problem for macrocells, (15d) can be easily converted into a linear constraint. Therefore, the optimization problem (15) is a linear-constrained optimization problem. However, the objective function of the optimization problem (15) is more challenging than the optimization problem (13) of macrocells because there is more than one variable in the numerator of γ_u . Let $I_u(\mathbf{P}_{\mathcal{C}}) = I_{u,\mathcal{F}}^{(n)} + I_{u,\mathcal{C} \setminus \mathcal{C}_i}^{(n)}$. To find the lower bound of $\varphi_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}})$, we use inequality (A3) in Appendix A.2. In detail, for $x_{u,b} = |h_{u,b}^{(n)}|^2 p_{u,b}^{(n)}$, $y_u = I_u(\mathbf{P}_{\mathcal{C}}) + \sigma_u^2$, and at the κ th iteration $\bar{x}_{u,b} = |h_{u,b}^{(n)}|^2 p_{u,b}^{(n(\kappa))}$, $\bar{y}_u = I_u(\mathbf{P}_{\mathcal{C}}^{(\kappa)}) + \sigma_u^2$, where $\mathbf{P}_{\mathcal{C}}^{(\kappa)}$ is a feasible point of the optimization problem (15). Following inequality (A3), we have $\varphi_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}}) \geq \tilde{\varphi}_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}}, \mathbf{P}_{\mathcal{C}}^{(\kappa)})$, where $\tilde{\varphi}_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}}, \mathbf{P}_{\mathcal{C}}^{(\kappa)}) \triangleq \sum_{u=1}^U \left(\tilde{a}_u^{(\kappa)} - \sum_{b \in \mathcal{C}_i} \tilde{b}_{u,b}^{(\kappa)} / x_{u,b} - \tilde{c}_u^{(\kappa)} y_u \right)$ and $\bar{s}_u = \sum_{b \in \mathcal{C}_i} \bar{x}_{u,b}$, $\tilde{a}_u^{(\kappa)} = \log_2 \left(1 + \frac{\bar{s}_u}{\bar{y}_u} \right) + \frac{2\bar{s}_u}{\ln(2)(\bar{s}_u + \bar{y}_u)}$, $\tilde{b}_{u,b}^{(\kappa)} = \frac{\bar{x}_{u,b}^2}{\ln(2)(\bar{s}_u + \bar{y}_u)}$, $\tilde{c}_u^{(\kappa)} = \frac{\bar{s}_u}{\ln(2)\bar{y}_u(\bar{s}_u + \bar{y}_u)}$.

Let $\lambda_{\mathcal{U}}^{(\kappa)} \triangleq \varphi_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}}^{(\kappa)}) / \pi_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}}^{(\kappa)})$. According to the Dinkelbach algorithm, the suboptimization problem to find $\mathbf{P}_{\mathcal{C}}^{(\kappa+1)}$ for iteration $(\kappa + 1)$ is as follows

$$\max_{p_{u,b}^{(n)}} \tilde{\varphi}_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}}, \mathbf{P}_{\mathcal{C}}^{(\kappa)}) - \lambda_{\mathcal{U}}^{(\kappa)} \pi_{\mathcal{U}}(\mathbf{P}_{\mathcal{C}}) \quad \text{s.t.} \quad (15b), (15c), (15d). \quad (16)$$

The same as for optimization problem (14), optimization problem (16) is also a convex problem. The proof of convergence is presented in Appendix B.2. Figure 3 shows a diagram for solving the centralized PA optimization problems, with ϵ denoting the tolerance value and $n_{\max}$ being the maximum number of iterations. If the values of the objective functions in both (14) and (16) are lower than the tolerance value ϵ or $n_{lp} \geq n_{\max}$, with n_{lp} denoting the number of iterations, then the convergence holds true. With the number of decision variables BUN and the number of constraints $B + BUN + U$, the computational complexity of each iteration for solving (14) is $\mathcal{O}\left((BUN)^2(B + BUN + U)^{2.5} + (B + BUN + U)^{3.5}\right)$.

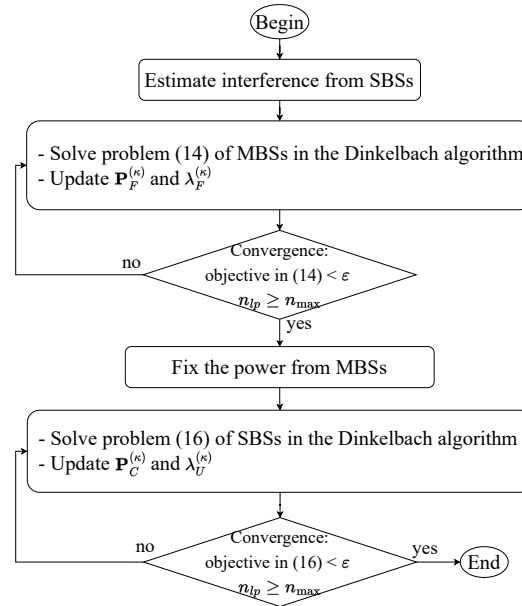


Figure 3. The flow chart of the proposed method to solve centralized PA.

5.2. Distributed Power Allocation by the Stackelberg Game

The solution of centralized PA methods is the best option for PA to MBSs and SCDs. However, when the number of small cells increases, the execution time of these methods also significantly rises. Therefore, centralized PA methods cannot satisfy the strict time constraints of UDNs in B5G. In this subsection, we propose a distributed PA method based on a SG to considerably reduce the complexity of the original optimization problems while retaining an acceptable efficiency of the solution. A SG is a type of NCG where players have no incentive to collaborate. There are two levels of priority in a SG. In detail, players with a high level of priority are called leaders and the others are followers. When starting a SG, leaders choose optimal actions first. Then, followers see the actions of the leaders to choose their optimal actions. Thus, a SG is also called a two-stage NCG.

The distance from MBSs to MUEs is usually much larger than the distance between SCDs to SUEs. Therefore, the problems of reflection, diffraction, and scattering influence the received signal at MUEs more than at SUEs. Meanwhile, MBSs must guarantee serving all UEs that are out of the coverage of the small cells, with the constraint of data rate. Therefore, MBSs, as leaders, have a high level of priority, and SCDs are followers in this SG.

5.2.1. NCG for Leaders (MBSs)

We formulate the optimization problem (8) as an NCG, where $NCG_{leader} = \langle \mathbf{PL}_{\mathcal{F}}, \mathbf{P}_{\mathcal{F}}, ul_{\mathcal{F}} \rangle$, where $\mathbf{PL}_{\mathcal{F}} = \mathcal{F}$ is the set of players (i.e., MBSs), $\mathbf{P}_{\mathcal{F}}$ is a matrix of PA for all MBSs, and $ul_{\mathcal{F}}(f)$ is the utility function of the EE of MUEs served by MBS f . The solution to NCG_{leader} is a Nash equilibrium where no MBS has an incentive to change its power, and the power of each MBS is the best response to the power of other MBSs [30]. By using an NCG, the optimization problem (8) can be divided into multiple suboptimization problems for MBSs to obtain the best response for each player. In each subOP, we find an

optimal solution for an MBS while the power of the other MBSs is fixed. A subOP for MBS f is formulated as

$$\max_{\mathbf{P}_f} \quad ul_{\mathcal{F}}(f) = \frac{\sum_{m \in \mathcal{M}_f} \log_2(1 + \gamma_m)}{P_f^{total}} = \frac{A_f(\mathbf{P}_f)}{B_f(\mathbf{P}_f)} \quad (17a)$$

$$\text{s.t.} \quad \sum_{n=1}^N \sum_{m \in \mathcal{M}_f} a_{m,f}^{(n)} p_{m,f}^{(n)} \leq P_{MBS}^{\max}, \quad (17b)$$

$$p_{m,f}^{(n)} \geq 0, \forall m \in \mathcal{M}_f, n = 1, \dots, N, \quad (17c)$$

$$\gamma_m \geq \gamma_{m \min}, \forall m \in \mathcal{M}_f. \quad (17d)$$

The objective function (17a) is a concave–convex fraction. We can use the Dinkelbach method to solve the optimization problem (17). In detail, we introduce a new variable λ_f , and the optimization problem (17) can be rewritten as

$$\max_{\mathbf{P}_f} \quad UT_f(\mathbf{P}_f, \lambda_f) = A_f(\mathbf{P}_f) - \lambda_f B_f(\mathbf{P}_f) \quad \text{s.t.} \quad (17b), (17c), (17d). \quad (18)$$

The objective function in (18) is a concave function in the maximization problem, so the optimization problem (18) is a convex problem. Therefore, the optimization problem (18) can be effectively solved by using CVX or CVXPY [34,35]. The optimal solution $\mathbf{P}_f^*$ is also the optimal solution to optimization problem (18) if $UT_f(\mathbf{P}_f^*, \lambda_f^*) = 0$. We use Algorithm 3 to find the solution for the NCG NCG_{leader} . With the number of decision variables $M_f N$ and the number of constraints $1 + M_f N + M_f$, the computational complexity of each iteration for solving (18) is $\mathcal{O}\left((M_f N)^2(1 + M_f N + M_f)^{2.5} + (1 + M_f N + M_f)^{3.5}\right)$. It is clear that the complexity of (18) is much lower than that of (14) because $M_f < M$ and only one MBS is considered.

Algorithm 3 SG-based PA for macrocells and small cells.

- 1: Initialize power matrix $\mathbf{P}_{MBS}^{(0)}$ that meets constraints (17b), (17c), (17d), $k = 0$
 - 2: **while** the convergence is not reached
 - 3: $\mathbf{P}_{MBS}^{(k)} = \mathbf{P}_{MBS}^{(k-1)}$
 - 4: **for** $f \in \mathcal{F}$
 - 5: $\mathbf{P}_f^{(tmp)} = \mathbf{P}_f^{(k)}$
 - 6: **while** $UT_f(\mathbf{P}_f^{(tmp)}, \lambda_f) > \epsilon$
 - 7: $\lambda_f = A_f(\mathbf{P}_f^{(tmp)}) / B_f(\mathbf{P}_f^{(tmp)})$, $\mathbf{P}_f^{(tmp)} = \mathbf{P}_f^*$ where $\mathbf{P}_f^*$ is the solution of (18)
 - 8: $\mathbf{P}_f^{(out)} = \mathbf{P}_f^{(tmp)}$
 - 9: $\mathbf{P}_{MBS}^{(k)} = \mathbf{P}_{MBS}^{(out)}$, $k = k + 1$
 - 10: Initialize power matrix of SCDs $\mathbf{P}_{cluster}^{(0)}$ that meets constraints (19b), (19c), (19d), $k = 0$
 - 11: **while** the convergence is not reached
 - 12: $\mathbf{P}_{cluster}^{(k)} = \mathbf{P}_{cluster}^{(k-1)}$
 - 13: **for** $\mathcal{C}_i \in \mathcal{C}$
 - 14: $\mathbf{P}_{\mathcal{C}_i}^{(tmp)} = \mathbf{P}_{\mathcal{C}_i}^{(k)}$
 - 15: **while** $UT_{\mathcal{C}_i}(\mathbf{P}_{\mathcal{C}_i}^{(tmp)}, \lambda_{\mathcal{C}_i}) > \epsilon$
 - 16: $\lambda_{\mathcal{C}_i} = A(\mathbf{P}_{\mathcal{C}_i}^{(tmp)}) / B(\mathbf{P}_{\mathcal{C}_i}^{(tmp)})$, $\mathbf{P}_{\mathcal{C}_i}^{(tmp)} = \mathbf{P}_{\mathcal{C}_i}^*$ where $\mathbf{P}_{\mathcal{C}_i}^*$ is the solution of (20)
 - 17: $\mathbf{P}_{\mathcal{C}_i}^{(out)} = \mathbf{P}_{\mathcal{C}_i}^{(tmp)}$
 - 18: $\mathbf{P}_{cluster}^{(k)} = \mathbf{P}_{cluster}^{(out)}$, $k = k + 1$
-

5.2.2. NCG for Followers (Clusters)

We formulate optimization problem (9) as an NCG $NCG_{follower} = \langle \mathbf{PL}_C, \mathbf{P}_C, ul_C \rangle$, where $\mathbf{PL}_C = \mathcal{C}$ is the set of players (i.e., clusters), $\mathbf{P}_C$ is a matrix of PA for all clusters of SCDs, and $ul_C(\mathcal{C}_i)$ is the utility function of the EE of cluster $\mathcal{C}_i$. The PA problem for SCDs in (9) can be divided into multiple suboptimization problems by using NCG with players as clusters. A suboptimization problem of cluster $\mathcal{C}_i$ is as follows

$$\max_{\mathbf{P}_{\mathcal{C}_i}} ul_C(\mathcal{C}_i) = \frac{\sum_{u \in \mathcal{U}_{\mathcal{C}_i}} \log_2(1 + \gamma_u)}{\sum_{b \in \mathcal{C}_i} p_b^{\text{total}}} = \frac{A_{\mathcal{C}_i}(\mathbf{P}_{\mathcal{C}_i})}{B_{\mathcal{C}_i}(\mathbf{P}_{\mathcal{C}_i})} \quad (19a)$$

$$\text{s.t.} \quad \sum_{n=1}^N \sum_{u \in \mathcal{U}_{\mathcal{C}_i}} p_{u,b}^{(n)} \leq p_{\text{SCD}}^{\max}, \forall b \in \mathcal{C}_i, \quad (19b)$$

$$p_{u,b}^{(n)} \geq 0, \forall b \in \mathcal{C}_i, n = 1, \dots, N, \quad (19c)$$

$$\gamma_u \geq \gamma_{s \min}, \forall u \in \mathcal{U}_{\mathcal{C}_i}. \quad (19d)$$

To obtain the best response of cluster $\mathcal{C}_i$, the power of the other clusters is fixed. This best response is the solution to optimization problem (19). We use the Dinkelbach method to solve the optimization problem (19) because its objective function is a concave–convex fraction. In detail, we introduce a new variable $\lambda_{\mathcal{C}_i}$, and the optimization problem (19) can be rewritten as

$$\max_{\mathbf{P}_{\mathcal{C}_i}} UT_{\mathcal{C}_i}(\mathbf{P}_{\mathcal{C}_i}, \lambda_{\mathcal{C}_i}) = A_{\mathcal{C}_i}(\mathbf{P}_{\mathcal{C}_i}) - \lambda_{\mathcal{C}_i} B_{\mathcal{C}_i}(\mathbf{P}_{\mathcal{C}_i}) \quad \text{s.t.} \quad (19b), (19c), (19d). \quad (20)$$

The objective function in (20) is a concave function, so the optimization problem (20) is a convex optimization problem. Therefore, the optimization problem (20) can be effectively solved by using CVX or CVXPY [34,35]. The optimal solution $\mathbf{P}_{\mathcal{C}_i}^*$ is also the optimal solution to the optimization problem (20) if $UT_{\mathcal{C}_i}(\mathbf{P}_{\mathcal{C}_i}^*, \lambda_{\mathcal{C}_i}^*) = 0$. We use Algorithm 3 to find the solution to the NCG $NCG_{follower}$. With the number of decision variables $U_{\mathcal{C}_i} C_i N$ and the number of constraints $C_i + U_{\mathcal{C}_i} C_i N + U_{\mathcal{C}_i}$, the computational complexity of each iteration for solving (20) is $\mathcal{O}\left((U_{\mathcal{C}_i} C_i N)^2 (C_i + U_{\mathcal{C}_i} C_i N + U_{\mathcal{C}_i})^{2.5} + (C_i + U_{\mathcal{C}_i} C_i N + U_{\mathcal{C}_i})^{3.5}\right)$. The complexity of (20) is much lower than that of (16) since $U_{\mathcal{C}_i} < U$ and $C_i < B$.

6. Simulation Results

Next, we investigated the performance of the proposed methods in a two-tier UDN. Three hexagon macrocells with a coverage radius of 100 m were deployed next to each other. In each macrocell, one MBS equipped with 128 antennas was located in the center of the cell. In addition, many small cells that had a radius of 20 m were randomly distributed by PPP. A small cell had an SCD equipped with one omnidirectional antenna. The UEs in the coverage of any small cell were served by the cluster that contained that small cell, and the others were served by the MBSs. The parameters in the simulations are given in Table 3 [36]. Additionally, we introduced two more clustering methods for comparison. In the traditional clustering method, each SUE is served by the nearest SCD. Meanwhile, with random clustering, each SUE randomly connected to an SCD, which had a distance to this SUE of less than $d = 3 \times$ the radius of small cells. On the other hand, equal power allocation, where the power of $P_{\text{MBS}}^{\max}/2$ of each MBS is equally divided among its MUEs and the power $P_{\text{SCD}}^{\max}$ of each active SCD is equally divided among the subchannels, was also used for comparison with the EE maximization PA methods. To prove the performance improvement, the proposed methods were compared with conventional methods based on multiple aspects such as convergence speed of PA methods, energy efficiency, sum rate, and transmit power usage. The acronyms for the combinations of clustering and PA methods are given in Table 4. To solve the optimization problems and build simulating programs, CVXPY and PYTHON were used [35]. The computing platform was a PC with CPU@3.7 GHz and 32 GB RAM memory.

Table 3. Simulation parameters.

Parameter	Numerical Value
Carrier frequency/Total bandwidth	2 GHz/20 MHz
Bandwidth per subchannel	4 MHz
# subchannels	5
Path loss from MBSs to UEs	$128.1 + 37.6 \log_{10} d$ [dB], d in km
Path loss from SCDs to UEs	$140.7 + 36.7 \log_{10} d$ [dB], d in km
Shadowing standard deviation	8 dB
Noise power density	−174 dBm/Hz
Maximum transmit power of MBS	46 dBm
Maximum transmit power of SCD	30 dBm
Circuit power of MBS	$20\%P_{MBS}^{\max}$
Circuit power of SCD	$20\%P_{SCD}^{\max}$
The radius of a macrocell	100 m
The radius of a small cell	20 m
# transmit antennas per MBS	128
The minimum throughput per MUE	0.1 Mbps
The minimum throughput per SUE	1 Mbps

Table 4. ID of strategies for comparison.

ID	Clustering	PA for Macrocells	PA for Small Cells
CGCO	Coalition Game	Centralized optimization	Centralized optimization
CGDO	Coalition Game	Game-based optimization	Game-based optimization
TCDO	Traditional clustering	Game-based optimization	Game-based optimization
RCDO	Randomly clustering	Game-based optimization	Game-based optimization
TCEP	Traditional clustering	Equal power allocation	Equal power allocation

Figure 4 illustrates an example of a triple-macrocell UDN that combines 100 SUEs and 24 MUEs with a density of SCDs of 2000 SCDs/km². In the first close-up in Figure 4a, two adjacent small cells use two same subchannels to serve their SUEs. This causes high interference for these SUEs. Meanwhile, after using the coalition game, these two small cells form a cluster to cooperatively transmit (the first close-up in Figure 4b). Therefore, there is no interference in a cluster. Another advantage of clustering that is shown in the second close-up is to enhance the quality of the desired signal. The SUE (red) subchannel is served by two small cells that have almost the same distance to it instead of being served by a single cell. When the density of SCDs is very high, these situations occur many times. Thus, clustering SCDs based on coalition game, and cooperative transmission help to efficiently mitigate intercell interference and improve the power of the received signal.

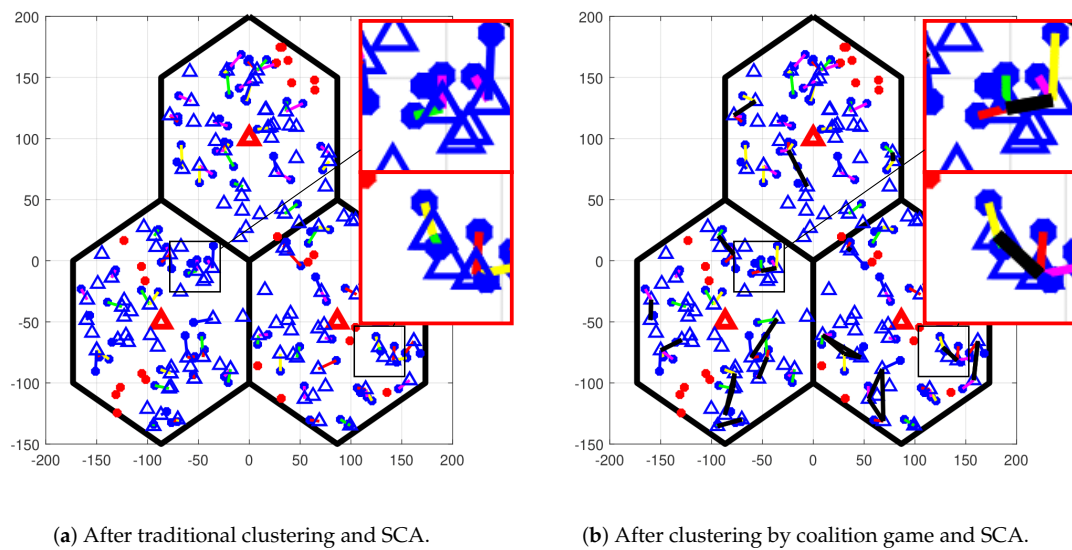


Figure 4. A typical triple-macrocell UDN after SCA with traditional and game-based clustering. Red triangles denote MBSs; red dots are MUEs; blue triangles are SCDs; blue dots are SUEs; SCDs in the same cluster are connected by black straight lines; SUEs served in the same subchannel are connected with their primal SCDs by the same color straight lines.

6.1. Convergence Speed of PA Methods

In this subsection, we consider the convergence of CGCO and CGDO in the network that has an SCD density of 2000 SCDs/km², 100 SUEs, 3 MBSs, and 24 MUEs. The changes and differences in the quantity of the EE of these methods, along with the iterations in the macrocell tier and small-cell tier, are shown in Figure 5a,b, respectively. It is given that one iteration of GT methods is complete after updating the best responses of all players. At the convergence, the EE of the CGDO is always lower than that of the CGCO. In detail, the EE of the CGDO is lower than that of CGCO in the macrocell tier and small-cell tier by 0.65% and 15.39%, respectively. However, the CGDO almost converges to the equilibrium after only one loop for updating all the best responses of the players, while it takes three and four iterations in the macrocell tier and small-cell tier, respectively, for the CGCO to achieve an EE of more than 95% at the convergence. On the other hand, in Figure 5b, the EE of the CGDO after one iteration is greater than the value of the convergence. This is reasonable because, in game theory, the value of the utility function at the Nash equilibrium may not be the best solution.

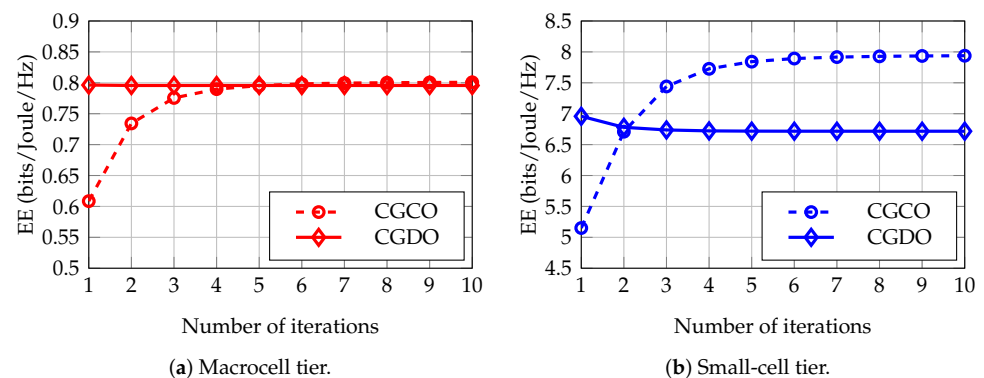


Figure 5. The convergence speed of the two PA methods.

6.2. Performance Analysis of CGCO, CGDO, and TCDO

We evaluated the performance of the proposed methods regarding the EE, the sum rate, the total transmit power, and the execution time in different scenarios with the number

of network elements that are given in Table 5, and the number of SUEs equaled 60% of the number of SCDs.

Table 5. The different network size scenarios.

Density (SCDs/km ²)	500	600	700	800	900	1000
# MBSs	3	3	3	3	3	3
# MUEs	9	12	15	18	21	24

Figure 6 displays data on the changes in the quantity of the EE of CGCO, CGDO, TCDO, RCDO, and TCEP in the small-cell tier when the density of SCDs increases from 500 to 1000 SCDs/km². To overview, the EE of all methods witnesses a decrease with different levels since the denser the deployment of small cells is, the higher interference is. The EE of CGCO is greatest in both the five methods since it uses the centralized PA method. In detail, the EE of CGDO equals 74.09% of the EE of CGCO on average. Meanwhile, the EE of CGDO is greater from 11.15% to 17.23% than the one of TCDO in all testing scenarios. In addition, CGCO, CGDO, and TCDO outperform RCDO and TCEP in terms of the EE. Therefore, using the coalition game for clustering small cells before SCA and PA based on distributed method helps improve the efficiency of energy usage in UDNs compared with the traditional clustering and the random clustering.

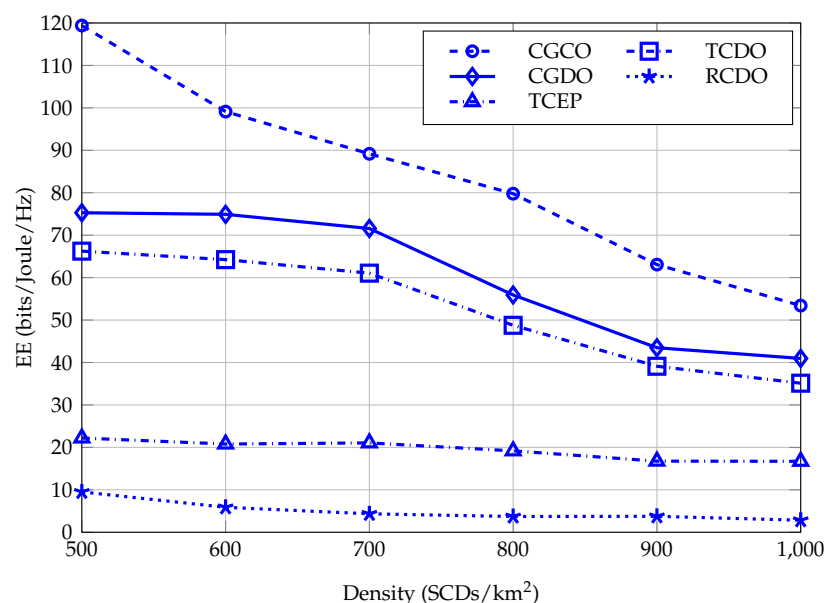


Figure 6. The changes in the EE of the methods in different scenarios.

Along with the EE, the computing execution time is an important factor to evaluate the performance of different methods. Table 6 describes the data in detail regarding the time required for the PA of CGCO, CGDO, and TCDO. Thanks to (12), the intensity of interference from small cells is pre-estimated in the PA optimization problems of the macrocell tier. Therefore, the execution time of the PA methods in the macrocell tier does not depend on the density of SCDs. On the other hand, the execution time for the PA of the MBSs of CGDO and TCDO is higher than the one of CGCO when the number of MUEs is small. The reason is that optimization problem (18) with the objective, a logarithmic function, takes a longer time to solve than optimization problem (16) with the objective as the sum of affine functions and rational functions when the number of variables is slightly different. However, the execution time in macrocells is less important than that in small cells with much more variables in the OPs. Considering the small-cell tier, when the density of the SCDs changes from 500 to 1000 SCDs/km², the execution time of the

CGCO significantly increases by 1.7 times, while those of the distributed methods CGDO and TCDO do not depend on the number of SCDs. Additionally, the execution time of CGCO is always longer, from 1.95 to 3.49 times that of CGDO. In large-scale UDNs, quick adaptation is key. If a very complex algorithm is used and the execution time is long, when the solution is obtained, it is not optimal anymore because the channel and user locations have changed too much. Therefore, the trade-off between EE and a quick execution time using our proposed PA method is essential and effective in UDNs.

Table 6. The average execution time for solving one optimization problem of PA methods for different network sizes.

Density (SCDs/km ²)	500	600	700	800	900	1000
CGCO MC (ms)	113	129	148	163	183	195
CGCO SC (ms)	243	284	323	364	379	411
CGDO MC (ms)	157	165	168	176	183	180
CGDO SC (ms)	125	124	124	122	115	118
TCDO MC (ms)	160	176	171	176	180	180
TCDO SC (ms)	118	112	114	109	103	104

Figure 7 shows the average of the sum rate and the total power consumption of CGCO, CGDO, and TCDO in the macrocell tier of UDNs. The level of the sum-rate growth of the three methods gradually decreases when the number of small cells increases in a certain area. The sum rate of these methods increases and reaches the maximum at a density of 900 SCDs/km². Then, it gradually decreases when densities are greater than 900 SCDs/km² despite the increased power usage because MUEs experience extreme interference from the small-cell tier. To clearly identify this trend, we added two more scenarios with densities of 1100 and 1200 SCDs/km², and we set the number of MUEs to 24. Additionally, in comparison with traditional clustering, SCDs were used in more subchannels for transmitting the signals to both their primal SUEs and the other SUEs in their clusters. This causes more interference from small cells to macrocells. Therefore, MBSS with CGDO use more power to achieve a higher sum rate than those of TCDO.

Figure 8 describes the sum rate and the total transmit power of the three methods, namely CGCO, CGDO, and TCDO, in the small-cell tiers of UDNs with different network sizes. When the density of the SCDs increases, the energy consumption considerably increases, while the rate of increase in the sum rate declines. Due to using centralized optimization, the data rate of CGCO is always larger than those of the others, while the total transmit power of CGCO is the lowest for the three methods. The power consumption of CGDO is lower by 9.51% to 15.19% than that of TCDO. On the other hand, when the deployment of SCDs becomes denser, to achieve the same data rate, the SCDs use much more power. For example, in Figure 8a, the sum rates, which CGDO and TCDO achieve with small cells with densities of 800 and 900 SCDs/km², are nearly equal. Meanwhile, the power consumption of CGDO and TCDO significantly increases in this range of density, as shown in Figure 8b. Moreover, the data rate of CGDO is slightly higher than that of TCDO. To summarize, the clustering method based on the coalition game helps to achieve a higher data rate than the traditional clustering method with lower power consumption.

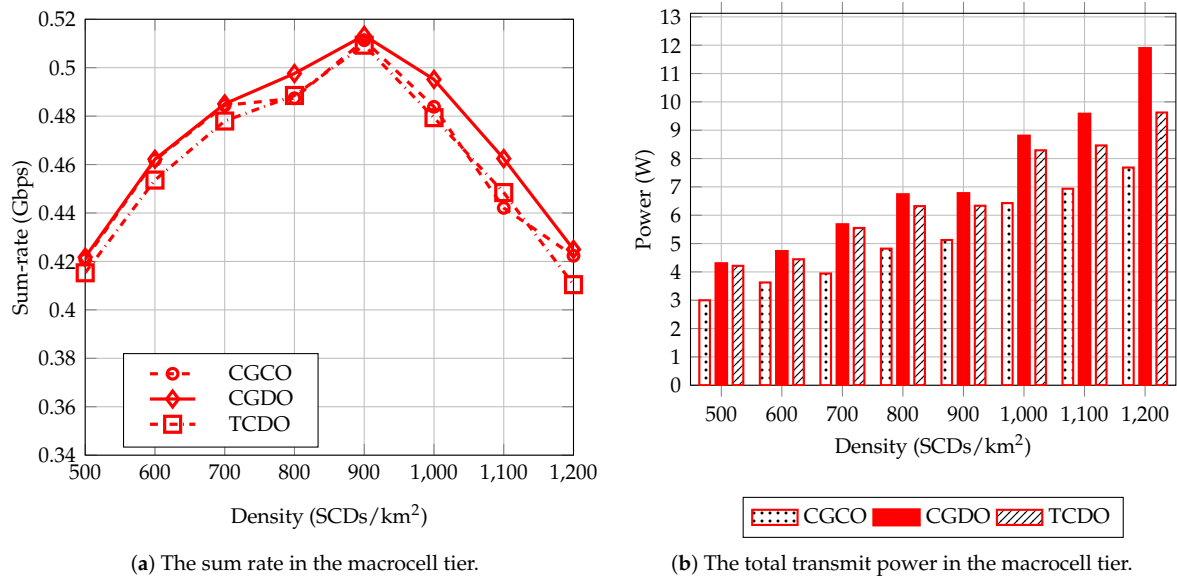


Figure 7. The sum rate and the total power consumption of the macrocells.

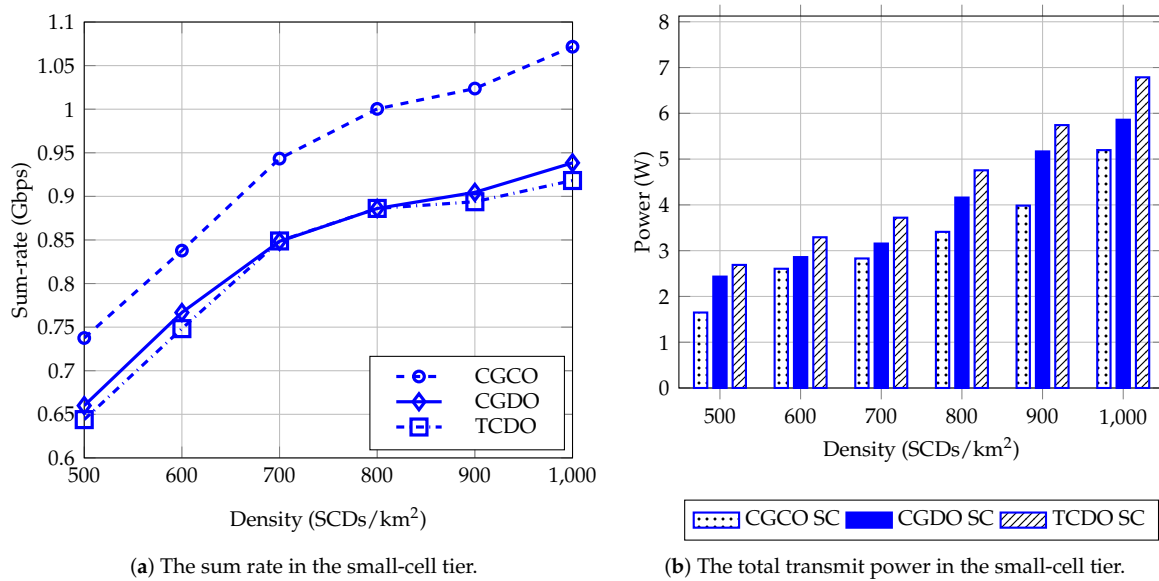


Figure 8. The sum rate and the total power consumption of the small cells.

6.3. Execution Time of Coalition Game (Algorithm 1) and the Hungarian Method (Algorithm 2)

Using the same simulation parameters as in Section 6.2, we evaluate the execution time of Algorithms 1 and 2 in Table 7. It is clear that the execution times of the coalition game and the Hungarian method increase with the increasing density of SCDs for the different levels. In detail, from 500 to 1000 SCDs/ km^2 , the execution time of the coalition game increases 3.8 times and that of the Hungarian method increases 3.1 times. The computational complexity of the Hungarian method is $O(N^4)$ in the worst case for the SCA in each cluster. Thus, its computational time has a linear relationship with the number of clusters in the small-cell tier.

Table 7. The execution time of the clustering-method-based coalition game and SCA-based Hungarian method for different network sizes.

Density (SCDs/km ²)	500	600	700	800	900	1000
Coalition game (μs)	282.2	309.1	426.7	543.4	586.9	1073.8
The Hungarian method (μs)	542.5	734.5	816.4	925.8	976.6	1688.6

7. Conclusions

We investigated the problem of resource allocation for UDNs with two tiers consisting of multiple MBSs and massive randomly distributed drones in this study. We proposed a paradigm that combines three processes: clustering using the coalition game, SCA using the Hungarian method, and PA using the SG as a distributed optimization method. Our simulations proved that the execution time of the centralized method for PA is much higher, by 1.95 to 3.49 times, than that of our distributed method, while the network EE of the proposed method using clustering based on the coalition game is improved by 11.15% to 17.23% compared with that of the distributed one with traditional clustering. Therefore, the centralized method is more suitable for UDNs with a small number of network elements, while the distributed method can be efficiently employed in large-scale UDNs where many MBSs and drones communicate simultaneously. Moreover, real-time computing plays an important role in drone-aided UDNs due to the mobility of both base stations (i.e., drones) and users. The execution time of the coalition game and the Hungarian method showed that they can be applied to real-time systems in UDNs.

For future work, with a massive number of users, data traffic is extremely high in drone-aided networks. To avoid network data congestion, drones equipped with caching storage, which can prestore common data packets, are a promising solution. However, the optimization problem for caching is integer programming with a large number of decision variables. Therefore, low-complexity algorithms for solving caching problems with the objective of minimizing the risk of network congestion are essential to be considered as an extension of this method. Additionally, to guarantee the quality of the signal received by users, drone-to-user communications need to be line of sight. Thus, the trajectory design for multiple drones in order to adapt to the movements of users is also an open topic to be addressed in the future.

Author Contributions: Conceptualization, T.T.B., L.D.N. and T.Q.D.; Methodology, T.T.B. and L.D.N.; Software, T.T.B.; Validation, L.D.N., H.H.K. and T.Q.D.; Formal analysis, T.T.B.; Investigation, L.D.N. and N.-S.V.; Resources, L.D.N. and H.H.K.; Writing—original draft, T.T.B. and L.D.N.; Writing—review & editing, H.H.K., N.-S.V. and T.Q.D.; Visualization, T.T.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Fundamental Inequalities

Appendix A.1

According to [32,37], we have

$$\log_2(1 + x/y) \geq \tilde{a} - \tilde{b}/x - \tilde{c}y, \quad (\text{A1})$$

where $\tilde{a} = \log_2(1 + \bar{x}/\bar{y}) + 2\bar{x}/(\ln(2)(\bar{x} + \bar{y})) > 0$, $\tilde{b} = \bar{x}^2/(\ln(2)(\bar{x} + \bar{y})) > 0$, $\tilde{c} = \bar{x}/(\ln(2)(\bar{x} + \bar{y})\bar{y}) > 0$.

Appendix A.2

Functions $h : \mathbb{R}^n \rightarrow \mathbb{R}$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $h(\mathbf{u}) \triangleq \log_2(\sum_{i=1}^n \exp(u_i))$ and $g_i(x_i) \triangleq \ln(1/x_i)$ are both convex in $\mathbf{x} = [x_i]_{i=1}^n > 0$ [38]. The extended-value extension $\tilde{h}$ of h is a nondecreasing function in each argument. Therefore, the vector composition of functions g and h is as follows

$$f(\mathbf{x}) \triangleq h(g(\mathbf{x})) = h(g_1(x_1), \dots, g_n(x_n)) = \log_2\left(\sum_{i=1}^n \exp(g_i(x_i))\right) = \log_2\left(\sum_{i=1}^n 1/x_i\right),$$

is convex. In addition, if we add a convex function to $f(\mathbf{x})$, then the convexity is preserved, and the new convex function is expressed as

$$f_1(\mathbf{x}, y) \triangleq f(\mathbf{x}) + \log_2\left(\frac{1}{y}\right) = \log_2\left(\sum_{i=1}^n \frac{1}{x_i}\right) + \log_2\left(\frac{1}{y}\right) = \log_2\left(\sum_{i=1}^n \frac{1}{x_i y}\right).$$

On the other hand, if z_0 is a positive real number, we have an inequality as follows [39]

$$\log_2(1 + z) \geq \alpha \log_2(z) + \beta, \quad (\text{A2})$$

where $\alpha = z_0/(1 + z_0)$, $\beta = \log_2(1 + z_0) - \alpha \log_2(z_0)$. The left part of (A2) is close to another part, and the equality holds true at $z = z_0$ [39].

Using (A2) and the convexity of $f_1(\mathbf{x}, y)$, we have

$$\begin{aligned} f_2(\mathbf{x}, y) &\triangleq \log_2\left(1 + \sum_{i=1}^n \frac{1}{x_i y}\right) \geq \bar{\alpha} \log_2\left(\sum_{i=1}^n \frac{1}{x_i y}\right) + \bar{\beta} \\ &\geq \bar{\alpha}(f_1(\bar{\mathbf{x}}, \bar{y}) + \langle \nabla f_1(\bar{\mathbf{x}}, \bar{y}), (\mathbf{x}, y) - (\bar{\mathbf{x}}, \bar{y}) \rangle) + \bar{\beta} \\ &= \underbrace{\bar{\alpha} f_1(\bar{\mathbf{x}}, \bar{y}) + \bar{\beta} + \bar{\alpha} \sum_{i=1}^n \frac{1}{\ln(2)\bar{y}\bar{x}_i \sum_{i=1}^n (1/(\bar{x}_i \bar{y}))}}_{\tilde{a}} + \bar{\alpha} \frac{\sum_{i=1}^n (1/\bar{x}_i)}{\ln(2)\bar{y} \sum_{i=1}^n (1/(\bar{x}_i \bar{y}))} \\ &\quad - \underbrace{\sum_{i=1}^n \frac{1}{\ln(2)\bar{y}\bar{x}_i^2 \sum_{i=1}^n (1/(\bar{x}_i \bar{y}))}}_{\tilde{b}_i} x_i - \underbrace{\frac{\sum_{i=1}^n (1/\bar{x}_i)}{\ln(2)\bar{y}^2 \sum_{i=1}^n (1/(\bar{x}_i \bar{y}))}}_{\tilde{c}} y, \end{aligned}$$

where $\bar{\alpha} = \left(\sum_{i=1}^n \frac{1}{\bar{x}_i \bar{y}}\right) / \left(1 + \sum_{i=1}^n \frac{1}{\bar{x}_i \bar{y}}\right)$, $\bar{\beta} = \log_2\left(1 + \sum_{i=1}^n \frac{1}{\bar{x}_i \bar{y}}\right) - \bar{\alpha} \log_2\left(\sum_{i=1}^n \frac{1}{\bar{x}_i \bar{y}}\right)$.

Replacing $x_i \rightarrow 1/x_i$ and $\bar{x}_i \rightarrow 1/\bar{x}_i$ in $f_2(\mathbf{x}, y)$, we have

$$\log_2\left(1 + \sum_{i=1}^n x_i/y\right) \geq \tilde{a} - \sum_{i=1}^n \tilde{b}_i/x_i - \tilde{c}y. \quad (\text{A3})$$

where $\tilde{a} = \log_2\left(1 + \frac{\bar{s}}{\bar{y}}\right) + \frac{2\bar{s}}{\ln(2)(\bar{s} + \bar{y})}$, $\tilde{b}_i = \frac{\bar{x}_i^2}{\ln(2)(\bar{s} + \bar{y})}$, $\tilde{c} = \frac{\bar{s}}{\ln(2)\bar{y}(\bar{s} + \bar{y})}$, $\bar{s} = \sum_{i=1}^n \bar{x}_i$.

Appendix B. Proving the Convergence

Appendix B.1

Let $\mathbf{P}_{\mathcal{F}}^{(\kappa+1)}$ be the solution of the convex optimization problem (14). Thus, the value of objective function in (14) at $\mathbf{P}_{\mathcal{F}}^{(\kappa+1)}$ is greater than the value of the other feasible points (i.e., $\mathbf{P}_{\mathcal{F}}^{(\kappa)}$) as [32]

$$\tilde{\varphi}_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa+1)}, \mathbf{P}_{\mathcal{F}}^{(\kappa)}) - \lambda_{\mathcal{M}}^{(\kappa)} \pi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa+1)}) > \tilde{\varphi}_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa)}, \mathbf{P}_{\mathcal{F}}^{(\kappa)}) - \lambda_{\mathcal{M}}^{(\kappa)} \pi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa)}). \quad (\text{A4})$$

From (A1), the equality is achieved when $x = \bar{x}$ and $y = \bar{y}$. Therefore, we have

$$\tilde{\varphi}_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa)}, \mathbf{P}_{\mathcal{F}}^{(\kappa)}) - \lambda_{\mathcal{M}}^{(\kappa)} \pi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa)}) = \varphi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa)}) - \lambda_{\mathcal{M}}^{(\kappa)} \pi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa)}) = 0. \quad (\text{A5})$$

With (A4) and (A5), we have

$$\lambda_{\mathcal{M}}^{(\kappa+1)} = \frac{\varphi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa+1)})}{\pi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa+1)})} \geq \frac{\tilde{\varphi}_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa+1)}, \mathbf{P}_{\mathcal{F}}^{(\kappa)})}{\pi_{\mathcal{M}}(\mathbf{P}_{\mathcal{F}}^{(\kappa+1)})} > \lambda_{\mathcal{M}}^{(\kappa)}. \quad (\text{A6})$$

Therefore, the first loop in Figure 3 converges at least to a local maximal point of the optimization problem (13).

Appendix B.2

Using the same approach as in Appendix II.1, we have $\lambda_{\mathcal{U}}^{(\kappa+1)} > \lambda_{\mathcal{U}}^{(\kappa)}$. From this, the second loop in Figure 3 also at least converges to a local maximal point of the optimization problem (15).

References

1. Cisco, V. Cisco Annual Internet Report (2018–2023) White Paper. 2020. Available online: <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf> (accessed on 10 October 2022).
2. Duong, T.Q.; Chu, X.; Suraweera, H.A. *Ultra-Dense Networks for 5G and Beyond: Modelling, Analysis, and Applications*; Wiley: Hoboken, NJ, USA, 2019.
3. Liu, J.; Zhang, Y.; Wang, J.; Cui, T.; Zhang, L.; Li, C.; Chen, K.; Li, S.; Feng, S.; Xie, D.; et al. Outage Probability Analysis for UAV-Aided Mobile Edge Computing Networks. *EAI Endorsed Trans. Ind. Netw. Intell. Syst.* **2022**, *9*, e4. [\[CrossRef\]](#)
4. Sharma, V. Functional Security and Trust in Ultra-Connected 6G Ecosystem. *EAI Endorsed Trans. Ind. Netw. Intell. Syst.* **2022**, *9*, e5. [\[CrossRef\]](#)
5. Cao, J.; Peng, T.; Liu, X.; Dong, W.; Duan, R.; Yuan, Y.; Wang, W.; Cui, S. Resource Allocation for Ultradense Networks With Machine-Learning-Based Interference Graph Construction. *IEEE Internet Things J.* **2020**, *7*, 2137–2151. [\[CrossRef\]](#)
6. Le, Q.N.; Nguyen, V.D.; Dobre, O.A.; Nguyen, N.P.; Zhao, R.; Chatzinotas, S. Learning-assisted user clustering in cell-free massive MIMO-NOMA networks. *IEEE Trans. Veh. Technol.* **2021**, *70*, 12872–12887. [\[CrossRef\]](#)
7. Lin, Y.; Zhang, R.; Yang, L.; Hanzo, L. Secure user-centric clustering for energy efficient ultra-dense networks: Design and optimization. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 1609–1621. [\[CrossRef\]](#)
8. Zhang, H.; Liu, H.; Jiang, C.; Chu, X.; Nallanathan, A.; Wen, X. A practical semidynamic clustering scheme using affinity propagation in cooperative picocells. *IEEE Trans. Veh. Technol.* **2015**, *64*, 4372–4377. [\[CrossRef\]](#)
9. Lin, Y.; Zhang, R.; Yang, L.; Hanzo, L. Modularity-based user-centric clustering and resource allocation for ultra dense networks. *IEEE Trans. Veh. Technol.* **2018**, *67*, 12457–12461. [\[CrossRef\]](#)
10. Nasir, A.A.; Tuan, H.D.; Duong, T.Q.; Poor, H.V. UAV-enabled communication using NOMA. *IEEE Trans. Commun.* **2019**, *67*, 5126–5138. [\[CrossRef\]](#)
11. Nguyen, M.H.T.; Garcia-Palacios, E.; Do-Duy, T.; Nguyen, L.D.; Mai, S.T.; Duong, T.Q. Spectrum-sharing UAV-assisted mission-critical communication: Learning-aided real-time optimisation. *IEEE Access* **2021**, *9*, 11622–11632. [\[CrossRef\]](#)
12. Masaracchia, A.; Li, Y.; Nguyen, K.K.; Yin, C.; Khosravirad, S.R.; Costa, D.B.D.; Duong, T.Q. UAV-enabled Ultra-Reliable Low-Latency Communications for 6G: A Comprehensive Survey. *IEEE Access* **2021**, *9*, 137338–137352. [\[CrossRef\]](#)
13. Nguyen, M.H.T.; Garcia-Palacios, E.; Do-Duy, T.; Dobre, O.A.; Duong, T.Q. UAV-aided aerial reconfigurable intelligent surface communications with massive MIMO system. *IEEE Trans. Cogn. Commun. Netw.* **2022**, *8*, 1828–1838. [\[CrossRef\]](#)
14. Duong, T.Q.; Kim, K.J.; Kaleem, Z.; Bui, M.P.; Vo, N.S. UAV caching in 6G networks: A Survey on models, techniques, and applications. *Phys. Commun.* **2022**, *51*, 101532. [\[CrossRef\]](#)

15. Mozaffari, M.; Saad, W.; Bennis, M.; Debbah, M. Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage. *IEEE Commun. Lett.* **2016**, *20*, 1647–1650. [[CrossRef](#)]
16. Gopi, S.P.; Magarini, M. Reinforcement learning aided UAV base station location optimization for rate maximization. *Electronics* **2021**, *10*, 2953. [[CrossRef](#)]
17. Chiaraviglio, L.; D'andreagiovanni, F.; Choo, R.; Cuomo, F.; Colonnese, S. Joint optimization of area throughput and grid-connected microgeneration in UAV-based mobile networks. *IEEE Access* **2019**, *7*, 69545–69558. [[CrossRef](#)]
18. Chiaraviglio, L.; D'Andreagiovanni, F.; Liu, W.; Gutierrez, J.A.; Blefari-Melazzi, N.; Choo, K.K.R.; Alouini, M.S. Multi-area throughput and energy optimization of UAV-aided cellular networks powered by solar panels and grid. *IEEE Trans. Mob. Comput.* **2021**, *20*, 2427–2444. [[CrossRef](#)]
19. Huang, D.; Cui, M.; Zhang, G.; Chu, X.; Lin, F. Trajectory optimization and resource allocation for UAV base stations under in-band backhaul constraint. *EURASIP J. Wirel. Commun. Netw.* **2020**, *2020*, 83. [[CrossRef](#)]
20. Yang, J.; Zhang, T.; Wu, X.; Liang, T.; Zhang, Q. Efficient Scheduling in Space-Air-Ground-Integrated Localization Networks. *IEEE Internet Things J.* **2022**, *9*, 17689–17704. [[CrossRef](#)]
21. Nguyen, M.N.; Nguyen, L.D.; Duong, T.Q.; Tuan, H.D. Real-time optimal resource allocation for embedded UAV communication systems. *IEEE Wirel. Commun. Lett.* **2019**, *8*, 225–228. [[CrossRef](#)]
22. Khodmi, A.; Rejeb, S.B.; Agoulmine, N.; Choukair, Z. A joint power allocation and user association based on non-cooperative game theory in an heterogeneous ultra-dense network. *IEEE Access* **2019**, *7*, 111790–111800. [[CrossRef](#)]
23. Wang, X.; Zhang, H.; Tian, Y.; Zhu, C.; Leung, V.C. Optimal distributed interference mitigation for small cell networks with non-orthogonal multiple access: A locally cooperative game. *IEEE Access* **2018**, *6*, 63107–63119. [[CrossRef](#)]
24. Xu, L.; Mao, Y.; Leng, S.; Qiao, G.; Zhao, Q. Energy-efficient resource allocation strategy in ultra dense small-cell networks: A Stackelberg game approach. In Proceedings of the IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–6.
25. Khodmi, A.; Rejeb, S.B.; Agoulmine, N.; Choukair, Z. Energy-Efficient Power Allocation and User-Channel Assignment for NOMA Heterogeneous Ultra-Dense Network. In Proceedings of the IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, London, UK, 31 August–3 September 2020; pp. 1–7.
26. Tinh, B.T.; Nguyen, L.D.; Kha, H.H.; Duong, T.Q. Joint Optimization of Clustering and Resource Allocation based on Game Theory for Ultra-Dense Networks. In Proceedings of the IEEE 9th International Conference on Communications and Electronics (ICCE), Nha Trang, Vietnam, 27–29 July 2022; pp. 23–28.
27. Huang, L.; Zhu, G.; Du, X. Cognitive femtocell networks: An opportunistic spectrum access for future indoor wireless coverage. *IEEE Wirel. Commun.* **2013**, *20*, 44–51. [[CrossRef](#)]
28. Saad, W.; Han, Z.; Debbah, M.; Hjørungnes, A.; Basar, T. Coalitional game theory for communication networks. *IEEE Signal Process. Mag.* **2009**, *26*, 77–97. [[CrossRef](#)]
29. Eliodorou, M.; Psomas, C.; Krikidis, I.; Socratous, S. User Association Coalition Games with Zero-Forcing Beamforming and NOMA. In Proceedings of the IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Cannes, France, 2–5 July 2019; pp. 1–5.
30. Tinh, B.T.; Nguyen, L.D.; Kha, H.H.; Duong, T.Q. Practical Optimization and Game Theory for 6G Ultra-Dense Networks: Overview and Research Challenges. *IEEE Access* **2022**, *10*, 2169–3536. [[CrossRef](#)]
31. Zhang, Z.; He, Y.; Chong, E. Opportunistic downlink scheduling for multiuser OFDM systems. In Proceedings of the IEEE Wireless Communications and Networking Conference, New Orleans, LA, USA, 13–17 March 2005; pp. 1206–1212.
32. Nguyen, L.D.; Tuan, H.D.; Duong, T.Q.; Poor, H.V.; Hanzo, L. Energy-Efficient Multi-Cell Massive MIMO Subject to Minimum User-Rate Constraints. *IEEE Trans. Commun.* **2021**, *69*, 914–928. [[CrossRef](#)]
33. Do-Duy, T.; Nguyen, L.D.; Duong, T.Q.; Khosravirad, S.; Claussen, H. Joint Optimisation of Real-time Deployment and Resource Allocation for UAV-Aided Disaster Emergency Communications. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 3411–3424. [[CrossRef](#)]
34. Grant, M.; Boyd, S. CVX: Matlab Software for Disciplined Convex Programming, Version 2.1. 2014. Available online: <http://cvxr.com/cvx> (accessed on 10 October 2022).
35. Diamond, S.; Boyd, S. CVXPY: A Python-embedded modeling language for convex optimization. *J. Mach. Learn. Res.* **2016**, *17*, 2909–2913.
36. Liu, G.; Sheng, M.; Wang, X.; Jiao, W.; Li, Y.; Li, J. Interference alignment for partially connected downlink MIMO heterogeneous networks. *IEEE Trans. Commun.* **2015**, *63*, 551–564. [[CrossRef](#)]
37. Sheng, Z.; Tuan, H.D.; Tam, H.H.M.; Nguyen, H.H.; Fang, Y. Energy-Efficient Precoding in Multicell Networks with Full-Duplex Base Stations. *EURASIP J. Wirel. Commun. Netw.* **2017**, *2017*, 48. [[CrossRef](#)]
38. Boyd, S.; Boyd, S.P.; Vandenberghe, L. *Convex Optimization*; Cambridge Univ. Press: Cambridge, UK, 2004.
39. Cheng, W.; Zhang, X.; Zhang, H. Optimal power allocation with statistical QoS provisioning for D2D and cellular communications over underlaying wireless networks. *IEEE J. Sel. Areas Commun.* **2016**, *34*, 151–162. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.