

RESEARCH

Open Access



# A low-complexity cell clustering algorithm in dense small cell networks

Ryuma Seno<sup>1\*</sup> , Tomoaki Ohtsuki<sup>1</sup>, Wenjie Jiang<sup>2</sup> and Yasushi Takatori<sup>2</sup>

## Abstract

Clustering plays an important role in constructing practical network systems. In this paper, we propose a novel clustering algorithm with low complexity for dense small cell networks, which is a promising deployment in next-generation wireless networking. Our algorithm is a *matrix-based* algorithm where metrics for the clustering process are represented as a matrix on which the clustering problem is represented as the maximization of elements. The proposed algorithm simplifies the exhaustive search for all possible clustering formations to the sequential selection of small cells, which significantly reduces the clustering process complexity. We evaluate the complexity and the achievable rate with the proposed algorithm and show that our algorithm achieves almost optimal performance, i.e., almost the same performance achieved by exhaustive search, while substantially reducing the clustering process complexity.

**Keywords:** Cell clustering algorithm, Small cell networks

## 1 Introduction

In the past few years, fifth generation (5G) wireless communication has become the center of discussion for researchers in this area [1, 2]. 5G is entirely different from conventional standards in that additional improvements cannot meet their requirements. In addition to the exponential increase in the data amount, the number of wireless devices connected to wireless networks should be considered. As represented by the term “Internet of Things” (IoT), it is predicted that many kinds of devices will operate on wireless networks, which will produce demand for much more network capacity [3].

A simple way to increase the network capacity is narrowing the range of cells and increasing the number of cells. Although this method is effective, inter-cell interference becomes more critical when densifying cells with small coverage [4], where radio resource management (RRM) would be of importance for interference mitigation. RRM schemes are mainly classified into two categories: frequency partitioning schemes and universal frequency reuse schemes. In the former, frequency resources that are orthogonal with each other are assigned

to each cell [5]. On the other hand, in the latter, many cells in the network share the same bandwidth [6]. These schemes may be combined into a hybrid scheme where the network is partitioned into a number of groups of cells and the cells in the same group share the same bandwidth. In [7], the network is classified into two areas, interference-sensitive area (ISA) and not-interference-sensitive area (NISA), and the entire bandwidth is partitioned into two parts, one where interference is tolerated and one where interference is prohibited, which results in the improved frequency utilization efficiency. In this scheme, how to partition the network greatly affects the performance achieved.

Ways for interference mitigation are also classified into inter-cell interference coordination (ICIC) and base station cooperation (BSC). In ICIC, neighboring cells use the orthogonal bandwidth and avoid the performance degradation of cell-edge users in each cell [8]. In [9], the dynamical management of bandwidth and cluster size results in improved frequency utilization efficiency. On the other hand, in BSC, the neighboring cells share the same bandwidth and cooperate with each other to make the interference beneficial [10]. Since a number of cells coordinate in BSC, the construction of groups in which cells coordinate greatly affects the performance. In both schemes, grouping cells that use the same bandwidth is essential to better performance.

\*Correspondence: [seno@ohtsuki.ics.keio.ac.jp](mailto:seno@ohtsuki.ics.keio.ac.jp)

<sup>1</sup>Graduate school of Science and Technology, Keio University, 3-14-1, Hiyoshi, Yokohama, Japan

Full list of author information is available at the end of the article

A particular example of a network deployment where inter-cell interference is notably critical is heterogeneous networks (small cell networks) [11]. Network operation by using only macro cells has a capacity hole problem where the achievable rate is extremely low. Small cell networks are attractive for addressing this problem because small cells can fill the capacity holes. Although small cells can enhance the network capacity, the dense deployment of cells results in severe inter-cell interference. The inter-cell interference problem in small cell networks has been addressed in many studies [12–15, 17, 19]. In [13], semi-distributed interference management in which cell clustering and resource allocation are jointly conducted was proposed for an orthogonal frequency division multiple access (OFDMA)-based two-tier cellular network. In [14], the authors proposed the joint optimization of small cell clustering and a beamforming vector at each base station in dense heterogeneous networks. In [15], interference alignment [16] was utilized as the interference management technique in heterogeneous networks where inter-tier and intra-tier interferences are eliminated by adequate beamforming and filtering. In [17], the authors proposed a scheme where some available degrees of freedom (DoF) of macro cells (primary users) were left and utilized for small cells (secondary users) in which interference alignment was achieved. This enabled the achievable DoF of small cells to be maximized while satisfying the required DoF of macro cells. In [18], clustering was performed on the basis of the rate loss caused by cells interfering with each other, where the sum of rate loss between cells belonging to different clusters is minimized, which means the rate loss caused by inter-cluster interference is minimized. In [19], the authors proposed clustering small cells in heterogeneous networks where interference alignment is utilized and found that clustering small cells is effective for improving spectral efficiency normalized with the number of antennas at a small cell base station.

The above discussion indicates that “clustering” plays an important role when operating a practical network system, and a clustering algorithm with lower complexity is desirable. Many kinds of clustering algorithms have been derived in the literature. A most simple clustering algorithm is an exhaustive search of all possible clustering patterns (exhaustive algorithm) [20]. Although an exhaustive algorithm can achieve global optimal performance, its complexity is significantly high, which is impractical. To achieve a practical clustering algorithm, many studies [21–23] have proposed schemes where each cluster is determined sequentially, i.e., a cluster is determined from cells that have not been selected yet. Papadogiannis et al. [21], Ng and Huang [22], and Qin and Tian [23] assume wireless networks adopting the coordination between BSs and consider the selection of coordinating

BSs. Papadogiannis et al. [21] proposes changing coordination sets dynamically to fully exploit the macro diversity, where each set is determined from BSs which are not selected yet so as to maximize the achievable rate in the set. Ng and Huang [22] consider wireless networks where each user receives the signal from several BSs and propose the cooperative precoding weight design. The set of coordinating BSs is determined so that the BSs within the set interfere with each other most strongly, i.e., each UE selects, at first, the geographically nearest BS and picks up some BSs to which the BS interferes most strongly, where the selected BSs are unified into a coordinating group. Qin and Tian [23] consider, as in [22], to group BSs interfering each other most strongly into a coordination set. In particular, the connectivity between BSs is modeled as a graph whose node corresponds to a BS and the weight of whose edge represents the level of the interference between two BSs and based on the graph the grouping is conducted. A partial graph holding the largest sum of weights among all possible sets, which becomes a coordination set, is separated from the whole graph, and this procedure is sequentially conducted until the grouping is completed. Note that all of these schemes have the feature that each BS cluster is determined sequentially one after another from BSs which are not selected yet. Such schemes are called “greedy algorithms” in general. Although such algorithms significantly reduce clustering process complexity, they necessitate sacrificing solution optimality.

In this paper, we propose a novel algorithm for cell clustering. Our algorithm is a *matrix-based* algorithm where metrics for the clustering process are represented as a matrix on which the clustering problem can be represented as the maximization of elements. In the proposed algorithm, we simplify the exhaustive algorithm into sequential selection of small cells through two-step transformations. The clustering problem is divided into some sub-problems each of whose objective function is represented as a sum of elements in the matrix, where each sub-problem corresponds to selecting a small cell. The transformations are conducted to narrow the search range in the same way as the transformation from an exhaustive algorithm to a greedy algorithm where solution optimality is replaced with complexity reduction. We evaluate the complexity and the achievable rate per small cell with the proposed algorithm and show that our algorithm achieves almost optimal performance, i.e., almost the same performance achieved in an exhaustive search, while significantly reducing the clustering process complexity.

The rest of the paper is organized as follows. Section 2 describes the system model and Section 3 describes the proposed algorithm. In Section 4, we present a performance evaluation in which we evaluated the clustering process complexity and the achievable rate per small cell.

Finally, Section 5 concludes the paper with a summary of key points.

## 2 System model

We consider downlink heterogeneous networks where many small cells coexist in a macro cell. The system model is depicted in Fig. 1.

In the macro cell, there are  $K$  small cells and  $K_{\text{muc}}$  macro user equipments (UEs), and in each small cell, there are  $K_{\text{sue}}$  small cell UEs. The macro base station (BS) and each small cell BS are respectively equipped with  $N_{\text{mbs}}$  and  $N_{\text{sbs}}$  antennas, and each UE is equipped with  $N_{\text{ue}}$  antennas. Each BS transmits  $d$  data streams to corresponding UE/UEs with spatial multiplexing. Note that in this paper, we assume there is only one macro cell because we focus on the clustering problem of small cells.

Small cells are divided into a number of groups. A clustering formation  $\mathbb{C}$  is denoted as follows.

$$\mathbb{C} = \{C_1, C_2, \dots, C_N\} \in \Omega \quad (1)$$

where  $C_n$  denotes the  $n$ th cluster,  $N$  denotes the number of clusters, and  $\Omega$  denotes the set of all clustering formations.  $C_n$  is further denoted as follows.

$$C_n = \{k_1^n, k_2^n, \dots, k_{|C_n|}^n\}, \quad k_l^n \in \mathcal{K} \quad (2)$$

where  $k_l^n$  denotes the index of the  $l$ th small cell in the  $n$ th cluster,  $|C_n|$  denotes the number of small cells in the  $n$ th cluster, and  $\mathcal{K}$  denotes the set of small cells. Note that the clustering formation is determined at a central unit to whom macro BS and each small cell BS connect via wired backhaul link. The required information (the selected clustering pattern, the channel information, etc.) are assumed to be exchanged via the backhaul link without delaying.

A metric is generally defined between cells in the case of cell clustering. There are a number of kinds of metrics, e.g., the distance between cells [22], the effect of multipath fading [24], and signal-to-interference-and-noise ratio (SINR) [25]. In this paper, we denote the metric between small cell  $i$  and small cell  $j$  by  $w_{ij}$  ( $\forall i, j \in \mathcal{K}$ ) and ignore the specific definition of the metric. Note that

we assume  $w_{ii} = 0$  and  $w_{ij} = w_{ji}$ ,  $\forall i, j \in \mathcal{K}$ . With the notation of  $w_{ij}$ , the sum of metrics in  $C_n$  and the sum of metrics within each cluster under  $\mathbb{C}$  are respectively represented as follows.

$$\mathcal{W}(C_n) = \sum_{ij \in C_n, i < j} w_{ij}, \quad (3)$$

$$\begin{aligned} \mathcal{U}(\mathbb{C}) &= \sum_{n=1}^N \mathcal{W}(C_n) \\ &= \sum_{n=1}^N \sum_{ij \in C_n, i < j} w_{ij}. \end{aligned} \quad (4)$$

In general, clustering problems are formalized as the maximization or minimization of the sum of metrics within each cluster. In this paper, without loss of generality, we formulate the clustering problem as follows.

$$\mathbb{C}^* = \arg \max_{\mathbb{C} \in \Omega} \mathcal{U}(\mathbb{C}) \quad (5)$$

where  $\mathbb{C}^*$  denotes the optimal clustering formation.

Although we focus on the clustering problem of small cells in this paper, the proposed algorithm can be easily applied to more general environments, e.g., multi-cell cellular networks or ad hoc sensor networks. In these cases, we only require that appropriate metrics are defined between cells or sensors, like  $w_{ij}$  which is defined between small cells in this paper.

The notations in this paper are summarized in Table 1.

## 3 Proposed method

In the proposed algorithm, metrics between small cells are represented as a matrix on the basis of which we transform the clustering problem. The clustering problem is re-formulated as the maximization of elements in the matrix, and we transform the exhaustive search of the optimal clustering pattern to the sequential selection of small cells.

### 3.1 Transformation of clustering process

We divide the metric between small cells  $w_{ij}$  as follows.

$$w_{ij} = \Delta_{ij} + \Delta_{ji} \quad (6)$$

where  $\Delta_{ij}$  denotes the effect of small cell  $j$  against small cell  $i$ , the SINR or the achievable rate loss, for example. For SINR,  $\Delta_{ij}$  indicates SINR at small cell  $i$  when interfered with by small cell  $j$ , and for the rate loss,  $\Delta_{ij}$  indicates the rate loss at small cell  $i$  when interfered with by small cell  $j$ . In the following, these effects are contained in  $\Delta_{ij}$ . Note that we assume  $\Delta_{ii} = 0$ ,  $\forall i \in \mathcal{K}$ . Although  $\Delta_{ij}$  should contain the effect of "all other small cells" against the small cell  $i$ , we emphasize that for the simplicity  $\Delta_{ij}$  contains the effect of "just the small cell  $j$ " against the small cell  $i$ .

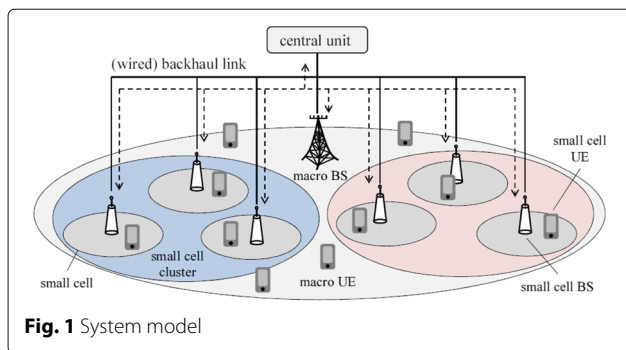


Fig. 1 System model

**Table 1** Notation

Symbol	Description
$K$	Number of small cells in a macro cell
$N$	Number of small cell clusters in a macro cell
$\mathbb{C}$	A clustering formation $\in \Omega$
$\Omega$	Set of clustering formations
$C_n$	$n$ th cluster
$k_l^n$	$l$ th small cell in $n$ th cluster
$\mathcal{K}$	Set of small cells
$w_{ij}$	Metric between small cell $i$ and small cell $j$
$\Delta_{ij}$	Effect from small cell $j$ to small cell $i$
$\mathcal{W}(C_n)$	Sum of metrics in $C_n$
$\mathcal{U}(\mathbb{C})$	Sum of metrics within each cluster under $\mathbb{C}$
$\Delta$	Matrix representing metrics
$\Delta(C_i)$	Sub-matrix corresponding to $C_i$ in $\Delta$
$\Theta$	Label based on which elements in $\Delta$ are arranged
$\mathcal{X}_l^n$	Set of small cells that have been selected By the $(l - 1)$ th cell in $C_n$
$\mathcal{Y}^n$	Set of small cells that are included in from $C_1$ to $C_n$
$\mathcal{Z}_l^n$	Set of small cells in $C_n$ up to $l$ th cell
$T^n$	Number of small cells that have been Selected from $C_1$ to $C_n$

We define the following matrix using  $\Delta_{i,j}$ .

$$\begin{aligned} \Delta &= \begin{bmatrix} \Delta_{1,1} & \Delta_{1,2} & \dots & \Delta_{1,K} \\ \Delta_{2,1} & \Delta_{2,2} & \dots & \Delta_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ \Delta_{K,1} & \Delta_{K,2} & \dots & \Delta_{K,K} \end{bmatrix} \\ &= \begin{bmatrix} 0 & \Delta_{1,2} & \dots & \Delta_{1,K} \\ \Delta_{2,1} & 0 & \dots & \Delta_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ \Delta_{K,1} & \Delta_{K,2} & \dots & 0 \end{bmatrix}. \end{aligned} \tag{7}$$

Note that the sum of elements being symmetrical with respect to the diagonal line is the metric between small cells, i.e.,  $\Delta_{i,j} + \Delta_{j,i} = w_{i,j}$ .

On the row and column of the matrix in Eq. (7), small cell indexes are labeled on the basis of which the elements are arranged, i.e., the element in the  $i$ th row and the  $j$ th column in  $\Delta$  is  $\Delta_{\theta_i, \theta_j}$  where  $\theta_i$  is the  $i$ th element in the label. Let  $\Theta$  denote the label. In Eq. (7), small cell indexes are labeled as follows.

$$\Theta = \{1, 2, \dots, K\} \tag{8}$$

which means that the  $i$ th row (and the  $i$ th column) is labeled by small cell  $i$ .

Applying a clustering formation  $\mathbb{C}$  results in the change of indexes in  $\Theta$ , which means the elements are rearranged on the basis of updated  $\Theta$  as in Eqs. (10) and (11) where “ $\times$ ” represents the elements that are not included in  $C_n$ ,  $\forall n$ . The updated label  $\Theta(\mathbb{C})$  is represented as follows.

$$\Theta(\mathbb{C}) = \left\{ \underbrace{k_1^1, k_2^1, \dots, k_{|C_1|}^1}_{C_1}, \underbrace{k_1^2, k_2^2, \dots, k_{|C_2|}^2}, \dots, \underbrace{k_1^N, k_2^N, \dots, k_{|C_N|}^N} \right\}. \tag{9}$$

$$\Delta(\mathbb{C}) = \begin{bmatrix} \Delta_{k_1^1, k_1^1} & \dots & \Delta_{k_1^1, k_{|C_1|}^1} & \dots & \Delta_{k_1^1, k_1^N} & \dots & \Delta_{k_1^1, k_{|C_N|}^N} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \Delta_{k_{|C_1|}^1, k_1^1} & \dots & \Delta_{k_{|C_1|}^1, k_{|C_1|}^1} & \dots & \Delta_{k_{|C_1|}^1, k_1^N} & \dots & \Delta_{k_{|C_1|}^1, k_{|C_N|}^N} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \Delta_{k_1^N, k_1^1} & \dots & \Delta_{k_1^N, k_{|C_1|}^1} & \dots & \Delta_{k_1^N, k_1^N} & \dots & \Delta_{k_1^N, k_{|C_N|}^N} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \Delta_{k_{|C_N|}^N, k_1^1} & \dots & \Delta_{k_{|C_N|}^N, k_{|C_1|}^1} & \dots & \Delta_{k_{|C_N|}^N, k_1^N} & \dots & \Delta_{k_{|C_N|}^N, k_{|C_N|}^N} \end{bmatrix} \tag{10}$$

$$= \begin{bmatrix} \Delta(C_1) & \times & \dots & \times \\ \times & \Delta(C_2) & \dots & \times \\ \vdots & \vdots & \ddots & \vdots \\ \times & \times & \dots & \Delta(C_N) \end{bmatrix} \tag{11}$$

Note that  $\Theta(\mathbb{C})$  represents the clustering formation  $\mathbb{C}$ . In the proposed algorithm, we select small cells sequentially from the first element of  $\Theta$  to the last one, which is equivalent to determining the clustering formation.

$\Delta(C_i)$  represents the matrix corresponding to  $C_i$  and is denoted as follows.

$$\Delta(C_i) = \begin{bmatrix} \Delta_{k_1^i, k_1^i} & \Delta_{k_1^i, k_2^i} & \dots & \Delta_{k_1^i, k_{|C_i|}^i} \\ \Delta_{k_2^i, k_1^i} & \Delta_{k_2^i, k_2^i} & \dots & \Delta_{k_2^i, k_{|C_i|}^i} \\ \vdots & \vdots & \ddots & \vdots \\ \Delta_{k_{|C_i|}^i, k_1^i} & \Delta_{k_{|C_i|}^i, k_2^i} & \dots & \Delta_{k_{|C_i|}^i, k_{|C_i|}^i} \end{bmatrix} \tag{12}$$

$$= \begin{bmatrix} 0 & \Delta_{k_1^i, k_2^i} & \dots & \Delta_{k_1^i, k_{|C_i|}^i} \\ \Delta_{k_2^i, k_1^i} & 0 & \dots & \Delta_{k_2^i, k_{|C_i|}^i} \\ \vdots & \vdots & \ddots & \vdots \\ \Delta_{k_{|C_i|}^i, k_1^i} & \Delta_{k_{|C_i|}^i, k_2^i} & \dots & 0 \end{bmatrix}. \tag{13}$$

Based on Eq. (7), the clustering process represented in Eq. (5) is transformed through a two-step transformation as follows.

- STEP 1 : exhaustive search for all possible clustering formations is transformed to the sequential

determination of each cluster in the same way as in a greedy algorithm

- STEP 2 : sequential determination of each cluster is transformed to the sequential selection of small cells

These transformations are conducted to narrow the search range, which results in reduced complexity. In particular, after the step 1 transformation, we determine each cluster sequentially instead of determining all clusters together, which means we search for small cells that have not been selected yet when determining a cluster. In addition, after the step 2 transformation, we sequentially determine each small cell instead of each cluster, which further reduces the complexity. By the sequential selection of small cells, we finally derive the clustering formation as represented in Eq. (9).

### 3.1.1 Step 1

The exhaustive search for all possible clustering formations is formalized as follows using Eqs. (3), (5), and (6).

$$\mathbb{C}^* = \arg \max_{\{C_1, C_2, \dots, C_N\} \in \Omega} \sum_{n=1}^N \sum_{i,j \in C_n} \Delta_{i,j} \quad (14)$$

where  $\sum_{i,j \in C_n} \Delta_{i,j}$  represents the sum of metrics in  $C_n$ .

Equation (14) is transformed as follows.

$$\arg \max_{\{C_1, C_2, \dots, C_N\} \in \Omega} \sum_{n=1}^N \sum_{i,j \in C_n} \Delta_{i,j} \quad (15)$$

$$\rightarrow \sum_{n=1}^N \arg \max_{C_n \subset \mathcal{K} \setminus \underbrace{\{C_1, C_2, \dots, C_{n-1}\}}_{\mathcal{Y}^{n-1}}} \sum_{i,j \in C_n} \Delta_{i,j} \quad (16)$$

where  $\mathcal{Y}^{n-1}$  denotes the set of small cells included in from  $C_1$  to  $C_{n-1}$ .

Equation (16) means that we determine each cluster sequentially, i.e., we search for small cells that have not been selected yet when determining a cluster according to the following equation.

$$\max_{C_n \subset \mathcal{K} \setminus \mathcal{Y}^{n-1}} \sum_{i,j \in C_n} \Delta_{i,j}, \quad n \in \{1, 2, \dots, N\}. \quad (17)$$

Equation (17) is called greedy algorithm in general.

### 3.1.2 Step 2

We divide the elements in  $\Delta(C_i)$  as follows.

$$\Delta(C_i) = \begin{bmatrix} 0 & \times & \star & \dots & \diamond \\ \times & \times & \star & \dots & \diamond \\ \star & \star & \star & \dots & \diamond \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \diamond & \diamond & \diamond & \dots & \diamond \end{bmatrix}, \quad \forall i \quad (18)$$

where each symbol represents the group of elements, i.e., the elements that have the same symbol are included in the same group. The division in Eq. (18) is represented mathematically as follows.

$$\sum_{i,j \in C_n} \Delta_{i,j} = \sum_{l=2}^{|C_n|} \sum_{i \in \underbrace{\{k_1^n, k_2^n, \dots, k_{l-1}^n\}}_{\mathcal{Z}_{l-1}^n}} \overbrace{(\Delta_{i,k_l^n} + \Delta_{k_l^n,i})}^{w_{i,k_l^n}} \quad (19)$$

where  $\mathcal{Z}_{l-1}^n$  denotes the set of small cells included in  $C_n$  up to the  $(l-1)$ th cell.

From Eqs. (19) and (17) is transformed as follows.

$$\arg \max_{C_n \subset \mathcal{K} \setminus \mathcal{Y}^{n-1}} \sum_{i,j \in C_n} \Delta_{i,j} \quad (20)$$

$$\rightarrow \arg \max_{C_n \subset \mathcal{K} \setminus \mathcal{Y}^{n-1}} \sum_{l=2}^L \sum_{i \in \mathcal{Z}_{l-1}^n} w_{i,k_l^n} \quad (21)$$

$$\rightarrow \sum_{l=2}^L \arg \max_{k_l^n \in \mathcal{K} \setminus \mathcal{X}_{l-1}^n} \sum_{i \in \mathcal{Z}_{l-1}^n} w_{i,k_l^n} \quad (22)$$

where  $\mathcal{X}_{l-1}^n$  is the set of small cells that have been selected by the  $(l-1)$ th cell in  $C_n$  and represented as follows.

$$\mathcal{X}_{l-1}^n = \underbrace{\{C_1, C_2, \dots, C_{n-1}\}}_{\mathcal{Y}^{n-1}} \underbrace{\{k_1^n, k_2^n, \dots, k_{l-1}^n\}}_{\mathcal{Z}_{l-1}^n}. \quad (23)$$

Therefore, when we determine small cells in  $C_n$ , we select small cells sequentially according to the following equation.

$$\max_{k \in \mathcal{K} \setminus \mathcal{X}_{l-1}^n} \sum_{i \in \mathcal{Z}_{l-1}^n} w_{i,k}, \quad l \in \{2, 3, \dots, |C_n|\}. \quad (24)$$

Note that Eq. (24) cannot determine the first small cell in each cluster. Hence, we follow the equation below when selecting the first small cell in each cluster.

$$k_1^n = \arg \max_{k \in \mathcal{K} \setminus \mathcal{Y}^{n-1}} \sum_{i \in \mathcal{K} \setminus \mathcal{Y}^{n-1}} w_{i,k}, \quad n \in \{1, 2, \dots, N\}. \quad (25)$$

Equation (25) means that for the first small cell in each cluster, we select the one that maximizes the sum of metrics between all small cells that have not been selected yet. Since we do not know which small cells are to be included in the same cluster when selecting the first small cell, it is better to select the one that maximizes the sum of metrics even if any small cells are selected as the same cluster, which corresponds to Eq. (25).

## 3.2 Proposed algorithm

The proposed algorithm is summarized in Algorithm 1.

**Algorithm 1** Proposed algorithm

---

```

1: INITIALIZE:  $\Delta, \Theta$ 
2: for  $n = 1, 2, \dots, N - 1$  do
3:   for  $l = 1, 2, \dots, |C_n|$  do
4:     if  $l = 1$  then
5:        $k^* = \arg \max_{k \in \mathcal{K} \setminus \mathcal{Y}^{n-1}} \sum_{i \in \mathcal{K} \setminus \mathcal{Y}^{n-1}} w_{i,k}$ 
6:     else
7:        $k^* = \arg \max_{k \in \mathcal{K} \setminus \mathcal{X}_{l-1}^n} \sum_{i \in \mathcal{Z}_{l-1}^n} w_{i,k}$ 
8:     end if
9:     set  $k^*$  to  $(T^{n-1} + l)$ th element of  $\Theta$  where  $T^n = \sum_{i=1}^n |C_i|$ 
10:  end for
11: end for

```

---

**4 Performance evaluation**

We evaluate the proposed algorithm in terms of two aspects: the complexity of clustering process and the achievable rate per small cell. As mentioned above, there is a trade-off between the complexity and the achievable rate, which means we have to sacrifice the optimality of the solution when reducing the complexity. However, we show that our algorithm is capable of achieving almost optimal performance, i.e., the same performance achieved by an exhaustive algorithm. Therefore, our algorithm can achieve almost optimal performance while reducing the complexity significantly.

In the following, we assume that each cluster comprises the same number of small cells, i.e., the following equation holds.

$$|C_n| = L, \quad \forall n \quad (26)$$

where  $L$  denotes a constant value that holds  $LN = K$ . Note that although there are some specific schemes in the literature [21–23], we exploit the essence of those schemes and refine them into the schemes described in the previous section, i.e., exhaustive algorithm and greedy algorithm, when comparing the performance of the proposed scheme, which is more concise.

**4.1 Complexity of clustering process**

All algorithms in this paper, i.e., exhaustive algorithm, greedy algorithm, and proposed algorithm have the objective function comprising only additions of metrics. Therefore, we use the total number of additions in the clustering process as the complexity. Note that the complexity to calculate the metrics themselves is independent of the clustering algorithm used, which allows us to ignore the complexity for the calculation of the metrics themselves. In the following, we derive the complexity of each algorithm quantitatively.

**4.1.1 Exhaustive algorithm**

Exhaustive algorithm is formalized in Eq. (14) where the objective function is given as Eq. (3). Hence, the number of additions in the objective function  $S_{\text{eh}}^1$  is expressed as follows.

$$\begin{aligned} S_{\text{eh}}^1 &= N \times \binom{L}{2} \\ &= \frac{NL(L-1)}{2} \end{aligned} \quad (27)$$

where the second term in Eq. (27) denotes the number of combinations of two small cells from  $C_n$ .

As represented in Eq. (14), the number of operations of the objective function in exhaustive algorithm  $S_{\text{eh}}^2$  is equal to the number of possible clustering formations, which is given as follows.

$$S_{\text{eh}}^2 = \frac{\prod_{i=0}^{N-2} \binom{L(N-i)}{L}}{N!}. \quad (28)$$

Therefore, the complexity of exhaustive algorithm  $S_{\text{eh}}$  is given as follows.

$$\begin{aligned} S_{\text{eh}} &= S_{\text{eh}}^1 \times S_{\text{eh}}^2 \\ &= \frac{NL(L-1)}{2} \cdot \frac{\prod_{i=0}^{N-2} \binom{L(N-i)}{L}}{N!}. \end{aligned} \quad (29)$$

From Eq. (29), the order of the complexity is represented as follows.

$$\mathcal{O}(S_{\text{eh}}) = \mathcal{O}(L^2 N^{L(N-1)+1}) \quad (30)$$

where the derivation is given in Appendix 1.

**4.1.2 Greedy algorithm**

Greedy algorithm is formulated as in Eq. (17) where the objective function is given in Eq. (3). Hence, the number of additions in the objective function  $S_{\text{gd}}^1$  is expressed as follows.

$$S_{\text{gd}}^1 = \frac{L(L-1)}{2}. \quad (31)$$

From Eq. (17), the number of operations of the objective function in greedy algorithm  $S_{\text{gd}}^1$  is given as follows.

$$S_{\text{gd}}^2 = \sum_{i=0}^{N-2} \binom{L(N-i)}{L}. \quad (32)$$

Therefore, the complexity of greedy algorithm is given as follows.

$$\begin{aligned} S_{\text{gd}} &= S_{\text{gd}}^1 \times S_{\text{gd}}^2 \\ &= \frac{L(L-1)}{2} \cdot \sum_{i=0}^{N-2} \binom{L(N-i)}{L}. \end{aligned} \quad (33)$$

From Eq. (33), we derive the order of the complexity as follows.

$$\mathcal{O} = \mathcal{O}(L^2 N^L) \tag{34}$$

where the derivation of Eq. (34) is given in Appendix 2.

#### 4.1.3 Proposal

The proposed algorithm is formalized as follows.

$$k_l^n = \arg \max_{k \in \mathcal{K} \setminus \mathcal{X}_{l-1}^n} \begin{cases} \sum_{i \in \mathcal{K} \setminus \mathcal{Y}^{n-1}} w_{i,k}, & l = 1 \\ \sum_{i \in \mathcal{Z}_{l-1}^n} w_{i,k}, & l \neq 1 \end{cases} \tag{35}$$

From Eq. (35), the number of additions in the objective function when selecting  $k_l^n$  is given as follows.

$$S_{pr}^1(n, l) = \begin{cases} lK - T^{(n-1)} - 1, & l = 1 \\ l - 1, & l \neq 1 \end{cases} \tag{36}$$

where  $T^{(n-1)}$  denotes the number of small cells selected from  $C_1$  to  $C_{n-1}$ , i.e.,  $T^{n-1} = L(n-1)$ .

The number of operations of the objective function in the proposed algorithm is given as follows.

$$S_{pr}^2(n, l) = K - T^{(n-1)} - (l - 1). \tag{37}$$

Therefore, the complexity of the proposed algorithm is given as follows.

$$\begin{aligned} S_{pr} &= \sum_{n=1}^N \sum_{l=1}^L (S_{pr}^1(n, l) \times S_{pr}^2(n, l)) \\ &= \frac{4}{3} N^3 L^2 - \frac{3}{2} N^2 L - \frac{3}{4} N^2 L^2 \\ &\quad + \frac{1}{4} N^2 L^3 + \frac{1}{2} NL - \frac{1}{12} NL^2 + \frac{1}{4} NL^3 \end{aligned} \tag{38}$$

where  $K = NL$ .

From Eq. (38), the order of the complexity is derived as follows.

$$\mathcal{O}(S_{pr}) = \mathcal{O}(N^3 L^2 + N^2 L^3) \tag{39}$$

#### 4.1.4 Comparison

We summarize the complexity of each algorithm in Table 2 and show the complexity of each algorithm when

**Table 2** Complexity of algorithms

Algorithm	Quantitative representation	Order representation
Exhaustive algorithm	Eq. (29)	$\mathcal{O}(L^2 N^{L(N-1)+1})$
Greedy algorithm	Eq. (33)	$\mathcal{O}(L^2 N^L)$
Proposal	Eq. (38)	$\mathcal{O}(N^3 L^2 + N^2 L^3)$

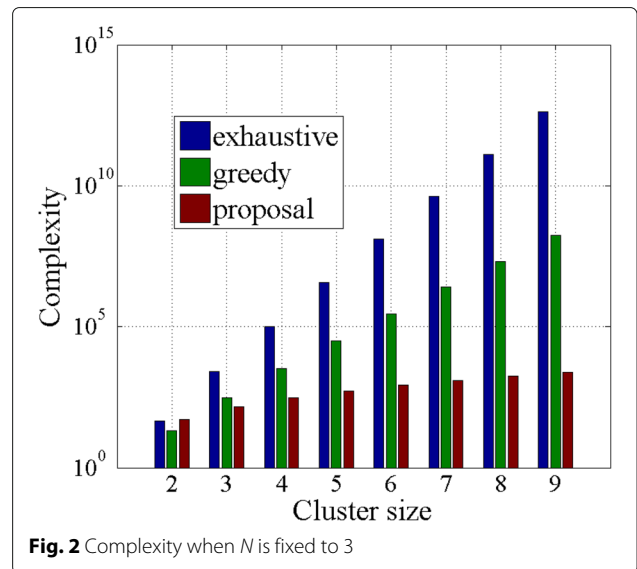
$N = 3$  in Fig. 2 and the complexity of each algorithm when  $L = 3$  in Fig. 3.

As shown in Table 2, the complexity of exhaustive algorithm increases exponentially with respect to both  $N$  and  $L$ , which can be verified in Figs. 2 and 3. In terms of greedy algorithm, as shown in Table 2, we found that the complexity increases exponentially with respect to  $L$  while polynomially with respect to  $N$ , which can be verified in Figs. 2 and 3.

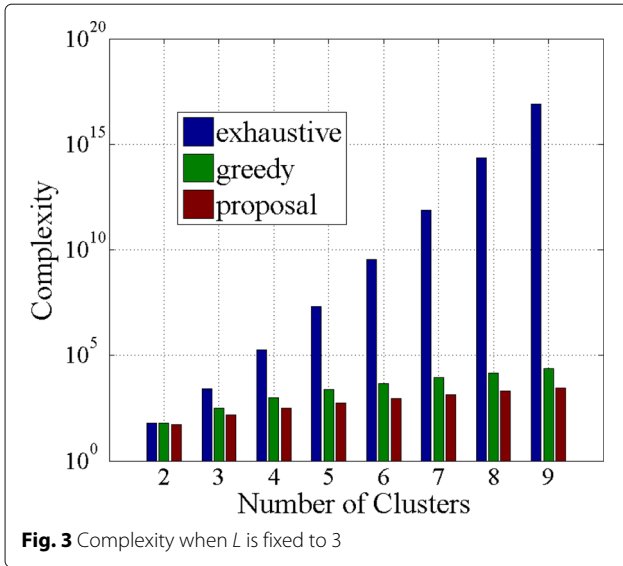
Compared to the aforementioned two algorithms, i.e., exhaustive algorithm and greedy algorithm, the complexity of the proposed algorithm increases polynomially with respect to both  $N$  and  $L$  as shown in Table 2. Therefore, our algorithm reduces the complexity significantly compared to the conventional algorithms, particularly when  $N$  or  $L$  increases.

#### 4.2 Achievable rate

In the work described in this paper, we conducted simulations following [19], i.e., precoding (postcoding) weight designs at BS (UE) and the simulation parameters were the same as [19]. The simulation parameters and weight designs are respectively shown in Table 3 and Table 4.



**Fig. 2** Complexity when  $N$  is fixed to 3



Small cells are uniformly deployed within the circle at whose center the macro BS is located, and each UE is uniformly deployed within the circle at whose center each corresponding BS is located. The simulation is conducted for each SNR on randomly generated 1000 channel conditions. As in [19], the metric between small cells, i.e.,  $w_{i,j}$ , is defined as follows.

$$w_{i,j} = \Delta_{i,j} + \Delta_{j,i} \tag{40}$$

where  $\Delta_{i,j}$  is the rate loss at small cell  $i$  when interfered with by small cell  $j$  and given as follows.

$$\begin{aligned} \Delta_{i,j} &= \log_2(1 + \rho_{ii}P_i) - \log_2\left(1 + \frac{\rho_{ii}P_i}{1 + \rho_{ij}P_j}\right) \\ &= \log_2\left(1 + \frac{\rho_{ii}P_i\rho_{ij}P_j}{1 + \rho_{ij}P_j + \rho_{ii}P_i}\right) \end{aligned} \tag{41}$$

**Table 3** Simulation parameters

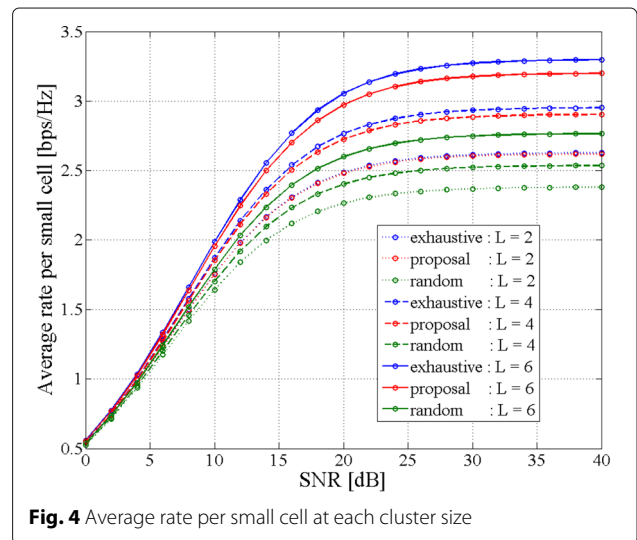
Macro cell	1		
Macro UE ( $K_{mue}$ )	12		
Small cell ( $K$ )	12		
Small cell UE ( $K_{sue}$ )	1 per small cell		
Macrocell radius	500 m		
Small cell radius	50 m		
Cluster size ( $L$ )	2	4	6
Transmit stream ( $d$ )	2		
Rx antenna ( $N_{ue}$ )	4		
Tx antenna			
macro BS ( $N_{mbs}$ )	48		
small BS ( $N_{sbs}$ )	4	12	20
Channel coefficient	i.i.d. Gaussian		
Path loss model	$-38.46 - \log_{10}(\text{distance (km)})$		

**Table 4** Design of weights

Type of weight	How to design	
Precoding weight	Macro BS	Direct the null space to macro UEs except for the desired UE and eliminate inter-stream interference
	Small BS	Align the intra-cluster interference to the strongest interference from macro BS
Postcoding weight	Macro UE	Eliminate inter-stream interference
	Small UE	ZF weight nulls out the aligned interference and MMSE weight mitigate the residual interference

where  $\sqrt{\rho_{ij}}$  and  $P_i$  respectively denote the path loss effect between small BS  $j$  and small UE  $i$  and the transmit power at small BS  $i$ . Note that, as in [19], we assume each small cell BS has the same transmit power, i.e.,  $P_i$  has the same value for all  $i$ , and we omit the effect of multipath fading in Eq. (41) for the simplicity.

Figure 4 shows the achievable rate per small cell for each cluster size. The  $x$ -axis and the  $y$ -axis respectively represent SNR and the achievable rate per small cell. Here, “exhaustive” represents the result achieved by exhaustive algorithm, “proposal” represents the result achieved by proposed algorithm, and “random” represents the result obtained when the clustering formation is determine randomly. As shown in the figure, “proposal” achieves almost the same rate as “exhaustive”, i.e., almost optimal rate. As mentioned before, since the proposed algorithm reduces the complexity compared to exhaustive algorithm, our algorithm is capable of achieving almost optimal performance while reducing the complexity significantly. In addition, it can be found that as the cluster size decreases, the difference between “exhaustive” and “proposal” also decreases. This is because as the cluster size becomes





smaller, the change in the clustering formation becomes less effective with respect to the rate performance.

Figure 5 shows the cumulative distribution function (CDF) of the achievable rate per small cell for each cluster size when SNR is 10 and 40 dB. The  $x$ -axis and the  $y$ -axis respectively represent the achievable rate per small cell and CDF. As shown in the figure, as SNR increases the variation of the achievable rate becomes larger. In fact, in the case that the cluster size is 6, although the achievable rate takes a value from 1.5 to 2.5 bps/Hz when SNR is 10 dB, it takes a value from 2 to 4 bps/Hz when SNR is 40 dB, where the range of the value becomes twice as large. Therefore, as SNR increases, the change of the clustering formation becomes more effective in the achievable rate.

Although there are some parameters in Table 3, only a few (“macro UE,” “small cell,” and “transmit stream”) can be set arbitrarily and the others should be set to a certain value(s) compulsorily due to the constraint regarding, mainly, the calculation of precoding or postcoding weight [19]. In the following, we investigate the performance when changing these parameters. Note that the cluster size is fixed to 4 unless otherwise specified.

*The number of macro UEs:* We assume that macro BS uses the given amount of transmit power to simultaneously serve macro UEs regardless of the number of macro UEs. Therefore, the total amount of the interference power that the macro BS exerts on small cells remains a certain level regardless of the number of macro UEs, which indicates even if we change the number of macro UEs, the achievable rate per small cell does not change. Fig. 6 shows the achievable rate per small cell versus SNR with different number of macro UEs. Note here that the number of small cells and the number of transmit streams are set to 12 and 2, respectively. As shown in the

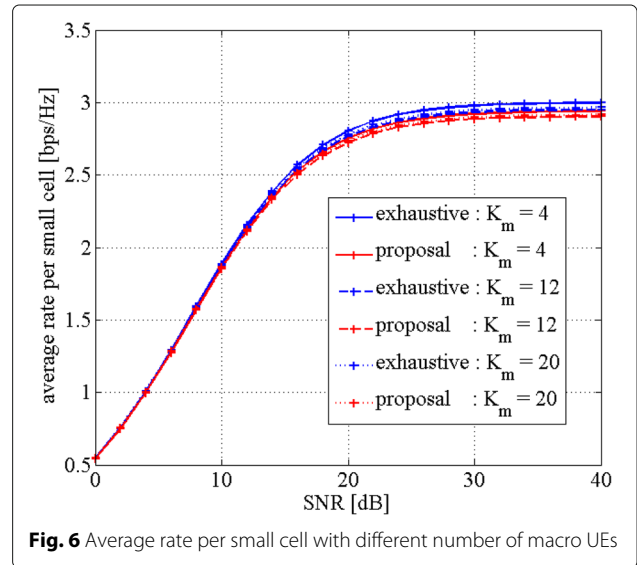


Fig. 6 Average rate per small cell with different number of macro UEs

figure, the achievable rate does not change regardless of the number of macro UEs. Note that in the figure  $K_m$  represents the number of macro UEs.

*The number of small cells:* With the cluster size fixed, increasing the number of small cells results in the severer inter-cluster interference because more small cells with fixed cluster size produce more clusters which interfere with each other. Therefore, it is easily expected that if we increase the number of small cells the achievable rate decreases and vice versa. Figure 7 shows the achievable rate per small cell versus SNR with different number of small cells where the number of small cells is represented as  $K$ . Note here that the number of macro UEs

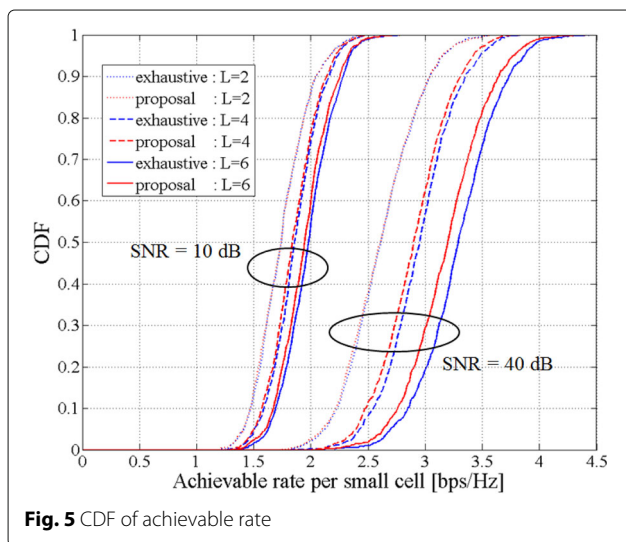


Fig. 5 CDF of achievable rate

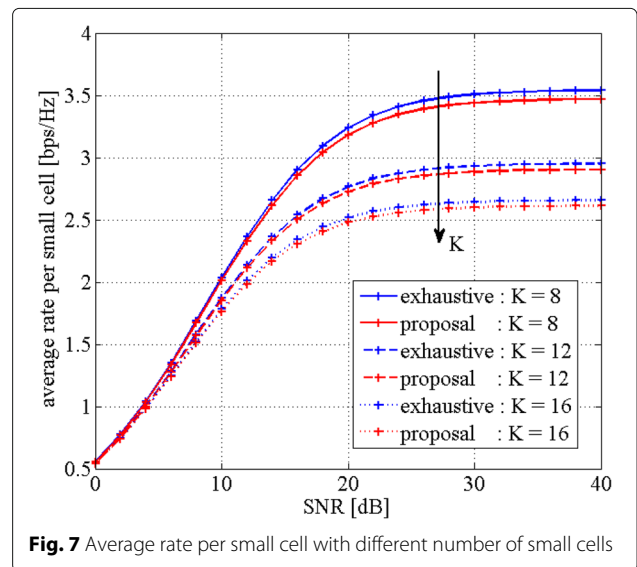
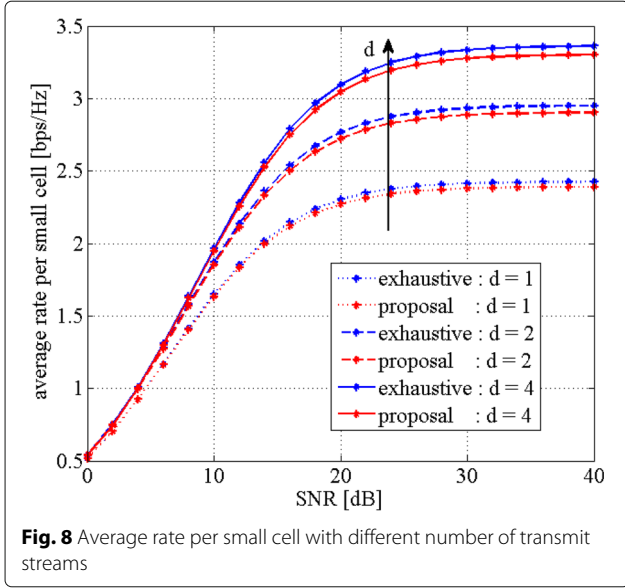


Fig. 7 Average rate per small cell with different number of small cells



and the number of transmit streams are set to 12 and 2, respectively. As shown in the figure, the achievable rate decreases as the number of small cells increases.

*The number of transmit streams:* Intuitively, increasing the number of transmit streams results in the increase of the achievable rate. Figure 8 shows the achievable rate per small cell versus SNR with different number of transmit streams. Note here that the number of macro UEs and the number of small cells are both set to 12. As shown in the figure, the achievable rate increases with the increase of the number of transmit streams.

### 5 Conclusions

In this paper, we proposed a novel cell clustering algorithm that achieves low complexity and high performance. Our algorithm is a *matrix-based* algorithm where the clustering problem is represented as the maximization of elements in a matrix representing the metric for the clustering process. The algorithm transforms an exhaustive search for all possible clustering formations to a sequential selection of small cells, which resulted in significantly reduced complexity. We evaluated the complexity and the achievable rate per small cell and showed that our algorithm achieved almost optimal performance, i.e., almost the same performance achieved by exhaustive search, while significantly reducing the clustering process complexity.

### Appendices

#### Appendix 1: the derivation of the complexity order for exhaustive algorithm

The complexity represented in Eq. (29) is transformed through Eqs. (42)–(48).

$$S_{eh} = \frac{NL(L-1)}{2} \cdot \frac{\prod_{i=0}^{N-2} \binom{L(N-i)}{L}}{N!} \quad (42)$$

$$= \frac{NL(L-1)}{2} \cdot \frac{\binom{LN}{L} \cdots \binom{L(N-i)}{L} \binom{L(N-(i+1))}{L} \cdots \binom{2L}{L}}{N!} \quad (43)$$

$$= \frac{NL(L-1)}{2N!} \quad (44)$$

$$\cdot \binom{LN}{L} \cdots \frac{L(N-i) \cdot (L(N-i)-1) \cdots \overbrace{(L(N-i)-(L-1))}^{L(N-(i+1))+1}}{L!} \times \frac{L(N-(i+1)) \cdot L(N-(i+1)-1) \cdots (N-(i+1)-(L-1))}{L!} \cdots \binom{2L}{L} \quad (45)$$

$$= \frac{NL(L-1)}{2N!} \quad (45)$$

$$\cdot \binom{LN}{L} \cdots \frac{L(N-i) \cdot (L(N-i)-1) \cdots (N-(i+1)-(L-1))}{(L!)^2} \cdots \binom{2L}{L} \quad (46)$$

$$= \frac{NL(L-1)}{2N!} \cdot \frac{LN \cdot (LN-1) \cdots (L+1)}{(L!)^{N-1}} \quad (46)$$

$$= \frac{NL(L-1)}{2N!} \cdot \frac{LN \cdot (LN-1) \cdots (L+1)}{(L!)^{N-1}} \cdot \frac{\overbrace{L \cdot (L-1) \cdots 1}^L}{L \cdot (L-1) \cdots 1} \quad (47)$$

$$= \frac{NL(L-1)}{2N!} \cdot \frac{(LN)!}{(L!)^N} \quad (48)$$

From Eq. (48), we derive the complexity order for exhaustive algorithm as follows.

$$\mathcal{O}(S_{eh}) = \mathcal{O}\left(\frac{NL(L-1)}{2N!} \cdot \frac{(LN)!}{(L!)^N}\right) \quad (49)$$

$$= \mathcal{O}\left(\frac{NL(L-1)}{2N^N} \cdot \frac{(LN)^{LN}}{(L^L)^N}\right) \quad (50)$$

$$= \mathcal{O}\left(\frac{NL(L-1)}{2} \cdot N^{L(N-1)}\right) \quad (51)$$

$$= \mathcal{O}\left(L^2 N^{(L(N-1)+1)}\right) \quad (52)$$

where we used the following relationship.

$$\mathcal{O}(X!) = \mathcal{O}(X^X). \quad (53)$$

#### Appendix 2: the derivation of the complexity order for greedy algorithm

The complexity order for greedy algorithm is derived as follows.

$$\mathcal{O}(S_{\text{gd}}) = \mathcal{O}\left(\frac{L(L-1)}{2} \cdot \sum_{i=0}^{N-2} \binom{L(N-i)}{L}\right) \quad (54)$$

$$= \mathcal{O}\left(\frac{L(L-1)}{2} \binom{LN}{L}\right) \quad (55)$$

$$= \mathcal{O}\left(\frac{L(L-1)}{2} \frac{LN \cdot (LN-1) \cdots (LN-(L-1))}{L!}\right) \quad (56)$$

$$= \mathcal{O}\left(\frac{L(L-1)}{2} \frac{LN \cdot (LN-1) \cdots (LN-(L-1))}{L^L}\right) \quad (57)$$

$$= \mathcal{O}\left(\frac{L(L-1)}{2} \cdot N \left(N - \frac{1}{L}\right) \cdots \left(N - \frac{L-1}{L}\right)\right) \quad (58)$$

$$= \mathcal{O}(L^2 N^L) \quad (59)$$

where we used the relationship shown in Eq. (53).

#### Competing interests

The authors declare that they have no competing interests.

#### Author details

<sup>1</sup>Graduate school of Science and Technology, Keio University, 3-14-1, Hiyoshi, Yokohama, Japan. <sup>2</sup>NTT Network Innovation Laboratories, 1-1, Hikarinooka, Yokosuka, Japan.

Received: 27 April 2016 Accepted: 31 October 2016

Published online: 14 November 2016

#### References

- JG Andrews, S Buzzi, W Choi, SV Hanly, A Lozano, ACK Soong, JC Zhang, What will 5G be? *IEEE J. on Sel. Areas in Commun.* **32**(6), 1065–1082 (2014)
- B Clerckx, A Lozano, S Sesia, 3GPP LTE and LTE-Advanced. *EURASIP J. on Wireless Commun. Netw.* **2009**, 472124 (2009). doi:10.1155/2009/472124
- A Zanella, N Bui, A Castellani, L Vangelista, M Zorzi, Internet of things for smart cities. *IEEE Internet of Things J.* **1**(1), 22–32 (2014)
- I Katzala, M Naghshineh, Channel assignment schemes for cellular mobile telecommunication systems: a comprehensive survey. *IEEE Pres. Commun.* **3**(3), 10–31 (1996)
- H-C Lee, D-C Oh, Y-H Lee, in *Proc. IEEE Int'l Conf. on Commun.* Mitigation of inter-femtocell interference with adaptive fractional frequency reuse (IEEE, 2010), pp. 1–5. doi:10.1109/ICC.2010.5502298
- X Chu, Y Wu, I Benmesbah, B Ling, in *Proc. IEEE Wireless Commun. Netw. Conf. Workshop.* Resource allocation in hybrid macro/femto networks (IEEE, 2010), pp. 1–5. doi:10.1109/WCNCW.2010.5487658
- H Li, X Xu, D Hu, X Tao, P Zhang, S Ci, H Tang, Clustering strategy based on graph method and power control for frequency resource management in femtocell and macrocell overlaid system. *IEEE J. of Commun. and Netw.* **13**(6), 664–677 (2011)
- G Fodor, in *Proc. IEEE/ACM Int'l Workshop on Quality of Service.* Performance analysis of reuse partitioning technique for OFDM based evolved UTRA (IEEE, 2006), pp. 112–120. doi:10.1109/IWQOS.2006.250457
- Performance analysis and simulation results of uplink ICIC. *Nokia Siemens Networks, 3GPP TSG RAN WG1 #51bis*, Spain, Jan 14–18 (2008)
- H Zhang, B Mehta, AF Molisch, J Zhang, H Dai, in *Proc. IEEE Int'l Conf. on Commun.* On the fundamentally asynchronous nature of interference in cooperative base station systems (IEEE, 2007), pp. 6073–6078. doi:10.1109/ICC.2007.1006
- V Chandrasekhar, JG Andrews, Femtocell networks: a survey. *IEEE Commun. Mag.* **46**(9), 59–67 (2008)
- N Saquib, E Hossain, L Bao, DI Kim, Interference management in OFDMA femtocell networks: issues and approaches. *IEEE Wireless Commun. Mag.* **19**(3), 86–95 (2012)
- A Abdelnasser, E Hossain, DI Kim, Clustering and resource allocation for dense femtocells in a two-tier cellular OFDMA network. *IEEE Trans. on Wireless Commun.* **13**(3), 1628–1641 (2014)
- M Hong, R Sun, H Baligh, Z-Q Luo, Joint base station clustering and beamforming design for partial coordinated transmission in heterogeneous networks. *IEEE J. on Sel. Areas in Commun.* **31**(2), 226–240 (2013)
- W Shin, W Noh, K Jang, H Choi, Hierarchical interference alignment for downlink heterogeneous networks. *IEEE Trans. on Wireless Commun.* **11**(12), 4549–4559 (2012)
- G Sridharan, Yu Wei, Degrees of freedom of MIMO cellular networks: decomposition and linear beamforming design. *IEEE Trans. on Inf. Theory.* **61**(6), 3339–3364 (2015)
- D Castanheira, A Silva, A Gameiro, Set optimization for efficient interference alignment in heterogeneous networks. *IEEE Trans. on Wireless Commun.* **13**(10), 5648–5660 (2014)
- S Chen, RS Cheng, Clustering for interference alignment in multiuser interference networks. *IEEE Trans. on Veh. Technol.* **63**(6), 2613–2624 (2014)
- R Seno, T Ohtsuki, J Wenjie, Y Takatori, in *Proc. IEEE Veh. Technol. Conf. fall.* Interference alignment in heterogeneous networks using pico cell clustering (IEEE, 2015), pp. 1–5. doi:10.1109/VTCSFall.2015.7390989
- A Papadogiannis, GC Alexandropoulos, in *Proc. IEEE Int'l Conf. on Fuzzy Sys.* A value of dynamic clustering of base stations for future wireless networks (IEEE, 2010), pp. 1–6. doi:10.1109/FUZZY.2010.5583933
- A Papadogiannis, D Gesbert, E Hardouin, in *Proc. IEEE Int'l Conf. on Commun.* A dynamic clustering approach in wireless networks with multi-cell cooperative processing (IEEE, 2008), pp. 4033–4037. doi:10.1109/ICC.2008.757
- CTK Ng, H Huang, Linear precoding in cooperative MIMO cellular networks with limited coordination clusters. *IEEE J. on Sel. Areas in Commun.* **28**(9), 1446–1454 (2010)
- C Qin, H Tian, in *Proc. IEEE Consumer Commun. and Netw. Conf.* A greedy dynamic clustering algorithm of joint transmission in dense small cell deployment (IEEE, 2014), pp. 629–634. doi:10.1109/CCNC.2014.6866638
- K Hosseini, H Dahrouj, R Adve, in *Proc. IEEE Global Commun. Conf.* Distributed clustering and interference management in two-tier networks (IEEE, 2012), pp. 4267–4272. doi:10.1109/GLOCOM.2012.6503788
- J-M Moon, D-H Cho, in *Proc. IEEE Consumer Commun. and Netw. Conf.* Formation of cooperative cluster for coordinated transmission in multi-cell wireless networks (IEEE, 2013), pp. 528–533. doi:10.1109/CCNC.2013.6488494

Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](http://springeropen.com)