# A Practical Dynamic Clustering Scheme Using Spectral Clustering in Ultra Dense Network

1st Yanxia Liang
*Shaanxi Key Laboratory of Information Communication Network and Security Xi'an University of Posts and Telecommunications name of organization*
Xi'an, China
email:liangyanxia201@xupt.edu.cn

2nd Xin Liu
*lSchool of Information Engineering y Xi'an Eurasia University*
Xi'an, China
email: liuxin@eurasia.edu

3rd Jing Jiang
*Shaanxi Key Laboratory of Information Communication Network and Security Xi'an University of Posts and Telecommunications name of organization*
Xi'an, China
email: jiangjing18@foxmail.com

4th Jianbo Du
*Shaanxi Key Laboratory of Information Communication Network and Security Xi'an University of Posts and Telecommunications name of organization*
Xi'an, China
email: dujianboo@163.com

6th Changyin Sun
*Shaanxi Key Laboratory of Information Communication Network and Security Xi'an University of Posts and Telecommunications name of organization*
Xi'an, China
email: changyin_sun@163.com

6th Yongbin Xie
*Shaanxi Key Laboratory of Information Communication Network and Security Xi'an University of Posts and Telecommunications name of organization*
Xi'an, China
email: xieyongbin007@163.com

***Abstract*— Coordinated Multiple-Point (CoMP) technology is one of the most important technologies in ultra-dense networks (UDN). Effective CoMP clustering algorithm can provide additional gain for system performance, such as higher system throughput and spectral efficiency (SE) for cell-edge users. This paper, a novel dynamic clustering scheme based on spectral clustering in graph theory for CoMP-user is presented to maximize system SE and cell-edge users' throughput. Our proposed scheme mainly consists of weight design, graph construction and spectral clustering that can achieve good performance and low complexity. Simulation results show that the proposed algorithm yields significant gains of SE and throughput and low running time compared with some existing clustering schemes.**

***Keywords*— *Ultra Dense Network, CoMP, spectral clustering, Virtual cell, system throughput, spectrum efficiency***

## I. Introduction

Communication system requires high capacity due to the explosive proliferation of mobile user equipment (UE)[1], which results in Ultra-Dense Networks (UDNs) supporting high-rate multimedia services[2]. A network could be called UDN in which the dense of Access Points (Aps) is higher than active UEs, or the dense of small cells (SCs) are more than 1000 cells/kms[3]. Many low-power SCs [4-5] are deployed in large quantities in UDN in order to accommodate higher frequencies (eg. millimeter wave) and obtain large capacity gains. However, densely-deployed SCs induce severe Inter-Cell Interference (ICI) with aggressive frequency reuse[6], and need proper energy management [7] . ICI is one of the key factors that limits the system capacity in UDN[8], as well as the quality-of-services (QoS) of cell-edge users.

As an effective scheme to mitigate interference, coordinated multiple-point (CoMP) technology, also known as Multi-cell cooperative processing (MCP), becomes a key technology for future UDN. CoMP utilizes radio access vitalization strategies to connect entities massively. Nevertheless, it is impossible to coordinate all the entities globally. Therefore, each user could be cooperatively served by a limited number of base stations (BSs). Thus, it is important for CoMP to adopt an efficient clustering scheme.

The existing clustering methods can be divided into three categories: static clustering, dynamic clustering and semi-dynamic clustering. The static scheme [9] is simple, but fixed clusters lead to unnecessary joint processing. different levels of interference can't be managed well by static scheme, which can only provide limited throughput gain. In this sense, dynamic clustering algorithm [10-12] is more flexible and practical. The dynamic clustering algorithm maximizes the system capacity with low complexity under the premise of completeness of CSI. The dynamic clustering scheme can mitigate cochannel interference (CCI) dynamically, but it needs low complexity to adapt to rapid signaling exchange. Semi-dynamic clustering [13] is a tradeoff between performance and complexity, and is widely researched. This paper challenges the dynamic clustering algorithm with low complexity.

Many cluster methods have been developed to mitigate interference in UDN. Reference [14] proposed an improved *k*-means algorithm to cluster SCs adaptively by selecting alternative SCs and Cluster initial centers and applying k-mean algorithm dynamically. Reference[15] put a user into a cluster who has the minimum weight value in potential interference matrix, in order to allocate subbands. The cluster method is based on the principle of proximity. In our previous work [16], we proposes a LBG clustering algorithm, in which users are clustered by a iterated process in which all users are regrouped constantly until the difference between two cores, got in two successive processes, of each cluster are very small.

These methods have resolved the clustering problem in UDN and brought higher spectrum efficiency (SE) or shorter running time. But most methods used clustering algorithm based on partition, including our early research[16]. Actually, if we take the points, BSs or users, as vertices, and take the relationship between these points, interference or distance or something you cared, as the edges, the clustering problem could be solved by graph-clustering algorithm. Graph

This work was supported in part by the National Natural Science Foundation of China under Grant 6187012068, in part by the Innovation Team Project of Shaanxi Province under Grant 2017KCT-30-02 , and in part by the Shaanxi STA International Cooperation and Exchanges Project under Grant 2017KW-011 and Grant 2019KW-005

clustering originated from the problem of Seven Bridges in Gunnysburg. It belongs to Graph theory and geometric topology initiated by Euler. Graph clustering was used to study the community structure [17], which brings convenience to people's life in the community, for example, social network can recommend friends. Graph is one kind of data structures[18], which can be used to represent many complex systems in reality.

In this paper, we propose a spectral-clustering based algorithm, which is one of the graph-clustering algorithms. We partition users into different cooperated clusters to maximize the system throughput and improve the cell-edge users' performance in UDN. The proposed scheme is proved to be effective, compared with existing *K*-means and LBG algorithm.

The rest of this paper is organized as follows. We introduce the system model in Section II. Then we present the procedures of clustering and propose our spectral clustering in Section III. In Section IV, the performance of the proposed algorithms is evaluated by simulations. Finally, Section V summarizes this paper.

## II. SYSTEM MODEL

The system model is based on the scene of UDN. There are several densely deployed APs and randomly distributed users in the network. Each AP can schedule multiple users at the same time. Each AP or user uses one transmitting or receiving antenna. $B$ represents the set of all APs in the system; and $L$ APs are grouped in each cell cluster. The $i$th cluster of APs is recorded as $B_i$, which schedule the set of users marked as $U_i$. The AP in each cell cluster shares the scheduling user data and cooperatively provides services. The model is shown in Fig.1.

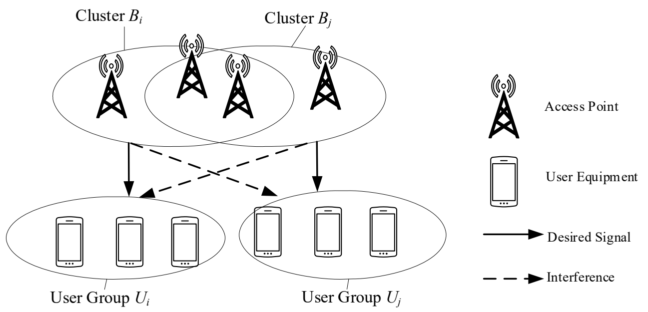


Fig. 1. System Model

Joint Transmission of CoMP is adopted in this paper. Multiple transmission nodes, APs in this model, provide services for the same user on the same time-frequency resource block. Users are taken into account first. From the perspective of a user, APs are clustered to serve for it. Different users who are served by same APs are grouped together too. This project mainly studies the clustering method of APs.

In our model, zero forcing algorithm precoding is utilized to eliminate the interference between users in a cluster. a round-robin scheduling scheme is applied, and the channel is assumed to be flat fading.

In non-CoMP system, each user has a main BS servicing for it. The cell-edge users are affected by serious cochannel interference (CCI) as shown in Fig. 2(a). In CoMP joint transmission, several APs constructing a cluster jointly transmit data, as shown in Fig.2(b).

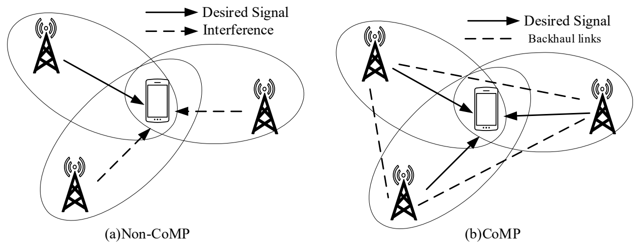


Fig. 2. Non-CoMP and CoMP system

In CoMP system, a central unit (CU) controls signals and data flow. We choose one of the APs in the cluster to be the master AP, and other APs act as slave Aps. All the Aps in a cluster are connected with each other by fibers to achieve backhaul transmission.

To simplify the analysis, we use $\boldsymbol{W_i}$ to represent zero-forcing precoding matrix at the user equipment (UE) devices and $u$ to denote the single CoMP user in cluster $B_v$. The received signal vector of user $v$ can be expressed as

$$\boldsymbol{y}_v = \underbrace{\boldsymbol{H}_v^{B_v}\boldsymbol{W}^{B_v}\boldsymbol{S}_v^{B_v}}_{\text{desired signal}} + \underbrace{\sum_{u\in U_u, u\neq v}\boldsymbol{H}_u^{B_u}\boldsymbol{W}^{B_u}\boldsymbol{S}_u^{B_u}}_{\text{intercluster interference}} + \boldsymbol{n}_v \quad , \tag{1}$$

where $\boldsymbol{H}_v^{B_v}$ represents the channel vector of user $v$ served by cluster $B_v$ ; $\boldsymbol{S}_v^{\boldsymbol{B}_v}$ represents the signal vector transmitted to user $v$ by all APs in cluster $B_v$, and each element is assumed with a power of $P_{tx}$ ; $\boldsymbol{n}_v$ represents the additive white Gaussian noise vector, $\boldsymbol{n}_v \sim CN\left(\boldsymbol{0},\sigma_n^2\boldsymbol{I}\right)$ ; $\boldsymbol{S}_v^{\boldsymbol{B}_v}$ , $\boldsymbol{y}_v$ and $\boldsymbol{n}_v$ are all in size of $|U_i|\times 1$.

The CoMP SINR for user $k$ in cluster $B_i$ is

$$SINR_B = \frac{\left|h_k^b\right| P_{Tx}}{\left|\sigma_n\right|^2 + \sum\limits_{u\in U_u, u\neq v}\left|h_u^b\right| P_{Tx}} . \tag{2}$$

In equation (2), $h_k^b$ is the channel parameter between user $k$ and AP $b$, which is an element of channel vector $\boldsymbol{H}_k^{B_i}$ . By using Shannon's capacity formula, the achievable capacity of user $k$ served by AP $b$ in CoMP is

$$C_k^b = B\log_2(1+SINR_B) . \tag{3}$$

In the $i$th AP cluster, the total capacity of user $k$ is $\sum\limits_{b\in B_i} C_k^b$ . So the aggregate capacity in $U_i$ is $\sum\limits_{k\in U_i}\sum\limits_{b\in B_i} C_k^b$ .
Supposed we've get the AP groups that served for some users, we focus on the proper user clustering method to maximize the system throughput, which is described as

$$U_i = \arg\max \sum_{i=1}^{L}\sum_{k\in U_i}\sum_{b\in B_i} C_k^b . \tag{4}$$

## III. Proposed Dynamic Clustering Scheme

Before getting into the clustering algorithm, two parts are considered. One is CoMP procedure, the other is the graph mapping of user clustering.

### *A. CoMP procedure*

User-centric CoMP is considered in the paper. From the respect of a user, several APs nearby are detected. Some of the APs are selected according to certain criterion associated with their locations and CSIs. More coordinating APs are needed when a user gets lower SINR, because it means that the user suffers greater interference. From the perspective of APs in a cluster, they share information and serve multiple users at the same time. At this point, we will face a problem how the users share the resources provided by APs with each other. In this paper, we aims at solving the problem by putting users in different clusters properly. The framework of CoMP procedure is shown in Fig.3.

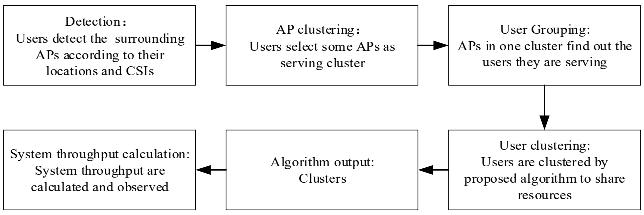


Fig. 3. Framework of CoMP procedure

### *B. Graph mapping of user clustering*

In non-CoMP scenario, cell-center users has stronger power than users at cell edge, who suffers more interference from other cells nearby. Thus they have poor performance, such as lower throughput. However, densely deployed cells make more cell-edge users. The system performance is affected strongly. In CoMP scenario, in order to increase the cell-edge users' throughput as well as the system throughput, we make as less interference as possible in one user cluster.

The observing users are served by a group of APs. We clustered users according to the interference between each other. The problem of user clustering can be turned into a graph cutting problem. First, we take users as vertices in a graph. The edges between vertices denote the relations between users. Interference is the most consideration between users. Parameters on the edge are defined as interference between users. There are directed graph and undirected graph. For simplicity, we use undirected graph. Fig. 4 shows sketch map of users and interference.

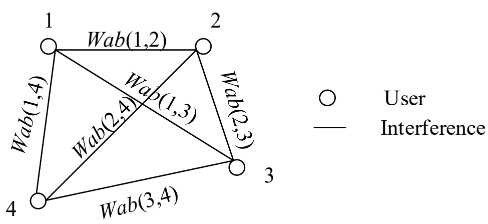


Fig. 4. Sketch map of users and interference

In Fig. 4, users are numbered. The interference between each two users is represented by *Wab*. *Wab*(*i*,*j*), called the weight, means the interference between user *i* and user *j*. At present, we have two new problems: one is how to get the *Wab*, the other is how to cut the graph properly. The latter will be detailed in part *C* in this segment. For the former one, by using cosine function in this part first.

We suppose that user $i$ and user $j$ are candidates in one cluster, and assume the composite channel vector as $H_{i,B_i}$ (composite channel vector of user $i$ in virtual cell [19-20] $B_i$), and $H_{j,B_j}$ (composite channel vector of user $j$ in virtual cell $B_j$). The weight between user $i$ in virtual cell $B_i$ and user $j$ in virtual cell $B_j$ is noted as

$$
\begin{aligned}
Wab(i,j) = & \left(1+\alpha\frac{\left|\boldsymbol{H}_{i,B_i}\boldsymbol{H}_{i,B_j}^{\mathrm{H}}\right|}{\left\|\boldsymbol{H}_{i,B_i}\right\|_{\mathrm{F}}\left\|\boldsymbol{H}_{i,B_j}\right\|_{\mathrm{F}}}\right)\cdot\left(1+\beta\frac{\left|\boldsymbol{H}_{i,B_i}\boldsymbol{H}_{j,B_i}^{\mathrm{H}}\right|}{\left\|\boldsymbol{H}_{i,B_i}\right\|_{\mathrm{F}}\left\|\boldsymbol{H}_{j,B_i}\right\|_{\mathrm{F}}}\right)\cdot\left\|\boldsymbol{H}_{i,B_i}\right\|_{\mathrm{F}}+, \\
& \left(1+\alpha\frac{\left|\boldsymbol{H}_{j,B_i}\boldsymbol{H}_{j,B_j}^{\mathrm{H}}\right|}{\left\|\boldsymbol{H}_{j,B_i}\right\|_{\mathrm{F}}\left\|\boldsymbol{H}_{j,B_j}\right\|_{\mathrm{F}}}\right)\cdot\left(1+\beta\frac{\left|\boldsymbol{H}_{i,B_j}\boldsymbol{H}_{j,B_j}^{\mathrm{H}}\right|}{\left\|\boldsymbol{H}_{i,B_j}\right\|_{\mathrm{F}}\left\|\boldsymbol{H}_{j,B_j}\right\|_{\mathrm{F}}}\right)\cdot\left\|\boldsymbol{H}_{j,B_j}\right\|_{\mathrm{F}}
\end{aligned}
\tag{5}
$$

where coefficients $\alpha$ and $\beta$ denote the proportions of power and space in multi-dimensional cooperation respectively. $\|\cdot\|_F$ represents the $F$ norm of the matrix, where $F$ is 2.

The nearer orthogonal the users' channels are, the less interference between users and the higher energy and frequency efficiency gains we get. In this case, weights are designed as follows in Table I.

TABLE I. Weights Design in Different Scenarios

| Scenarios | Weights |
|---|---|
| Channel vector cosine when user $i$ and user $j$ are in the same cell. | $\cos(\boldsymbol{H}_{i,B_i},\boldsymbol{H}_{j,B_i})=\frac{\lvert\boldsymbol{H}_{i,B_i}\boldsymbol{H}_{j,B_i}^{\mathrm{H}}\rvert}{\left\|\boldsymbol{H}_{i,B_i}\right\|_{\mathrm{F}}\left\|\boldsymbol{H}_{j,B_i}\right\|_{\mathrm{F}}}$ |
| Channel vector cosine when user $i$ and user $j$ are in different cells. | $\cos(\boldsymbol{H}_{i,B_i},\boldsymbol{H}_{i,B_j})=\frac{\lvert\boldsymbol{H}_{i,B_i}\boldsymbol{H}_{i,B_j}^{\mathrm{H}}\rvert}{\left\|\boldsymbol{H}_{i,B_i}\right\|_{\mathrm{F}}\left\|\boldsymbol{H}_{i,B_j}\right\|_{\mathrm{F}}}$ |

More details of the rationality of the weight design refers to our early work [21].

### *C. Proposed Spectral-based Clustering Scheme*

The second problem proposed in part B is how to cut the graph properly, which belongs to segmentation in graph theory. Spectral clustering[22] is a clustering algorithm based on spectral graph theory, which is easy to implement, and the effect of dividing nonlinear separable data is better than the traditional clustering algorithm.

In spectral clustering, all data are treated as points, which are connected by edges. The edge weight is low if it connects two long-distance points, while is high if it connects two short-distance points. The main idea of spectral clustering is to get the sum of edge weights between different subgraphs as low as possible by cutting the graph composed of all data points, while the sum of edge weights within the subgraph is as high as possible.

- Undirected weight graph Construction

We've designed a undirected weight graph in part *B* in this segment. The set of vertices (users) is expressed as *V* and the set of edges (weight *Wab*) as *E*. The graph is denoted as *G*(*V*,*E*). We get

$$V=\{V_1,V_2,\ldots,V_n\}, \tag{6}$$

where $V_i$ denote the user $i$. $Wab(i,j)$ is the weight between point $V_i$ and point $V_j$. $Wab(i,j)= Wab(j,i)$ due to undirected graph. For two points $V_i$ and $V_j$ with edge connection, $Wab(i,j) > 0$; without edge connection, $Wab(i,j)= 0$. This setting is consistent with the weight design.

For any point $V_i$ in a graph, its Degree $d_i$ is defined as the sum of the weights of all the edges connected to it, that is

$$d_i = \sum_{j=1}^{n} Wab(i,j) . \tag{7}$$

With the definition of Degree, we can get a degree matrix $\boldsymbol{D}$ in size of $n \times n$, which is a diagonal matrix. Its main diagonal values are the Degrees of the $i$th point in line $i$. Degree matrix is defined as

$$\boldsymbol{D} = \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n \end{bmatrix} \tag{8}$$

- Adjacency matrix construction

Using the weights between all points, we can get the adjacency matrix $\boldsymbol{W}$ in size of $n \times n$, whose $j$th value in line $i$ is weight $Wab(i,j)$. For example, the adjacency matrix $\boldsymbol{W}$ of Fig. 4 is

$$\boldsymbol{W} = \begin{bmatrix} Wab(1,1) & Wab(1,2) & Wab(1,3) & Wab(1,4) \\ Wab(2,1) & Wab(2,2) & Wab(2,3) & Wab(2,4) \\ Wab(3,1) & Wab(3,2) & Wab(3,3) & Wab(3,4) \\ Wab(4,1) & Wab(4,2) & Wab(4,3) & Wab(4,4) \end{bmatrix}, \tag{9}$$

which is a symmetric matrix because $Wab(i,j)= Wab(j,i)$. According to the usual principle: the edge weight is low if it connects two long-distance points, while is high if it connects two short-distance points. Interference between users(points) is low when two users are in long distance, and high when the two are in short distance. Therefore, it is suitable to use interference as the elements of adjacency matrix.

- Laplacian matrix and its properties

Laplacian matrix plays an important role in spectral clustering. Spectral clustering algorithm relies on Laplacian matrix and the properties of these matrices to get clustering results. We merely introduce the irregular Laplacian matrix. Degree matrix $\boldsymbol{D}$ minus adjacency matrix $\boldsymbol{W}$ is irregular Laplacian matrix $\boldsymbol{L}$, that is

$$\boldsymbol{L}= \boldsymbol{D} - \boldsymbol{W} . \tag{10}$$

Irregular Laplacian matrix $\boldsymbol{L}$ has the following properties[12]:

*1) For any vector* $\boldsymbol{f} = (f_1, \cdots, f_n)^{\mathrm{T}} \in \boldsymbol{R}^n$ ($\boldsymbol{R}^n$ denotes N-dimensional real space), we get

$$\boldsymbol{f}^{\mathrm{T}} \boldsymbol{L} \boldsymbol{f} = \frac{1}{2} \sum_{i,j=1}^{n} Wab(i,j)(f_i - f_j)^2 \tag{11}$$

*2) $\boldsymbol{L}$ is symmetric positive semidefinite.*

*3) The minimum eigenvalue of $\boldsymbol{L}$ is 0, and the corresponding eigenvector is $\boldsymbol{1}$.*

*4) $\boldsymbol{L}$ has n non-negative real eigenvalues, which satisfy* $0=\lambda_1 \le \lambda_2 \le \cdots \le \lambda_n$ .

These properties are proved in Ref. [12], and will not be discussed here.

The eigenvalues of irregular Laplacian matrices are also closely related to the number of connected subgraphs in undirected graphs.

From the assumption before, $G(V, E)$ is an undirected graph with nonnegative weight. There are $k$ connected subgraphs $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_k$ in the graph. The Laplace matrices of these connected subgraphs are $\{\boldsymbol{L}_i\}^k_{i=1}$. Without losing generality, $\boldsymbol{L}$ can be expressed in the form of block diagonal matrix as follows:

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{L}_1 & & & \\ & \boldsymbol{L}_2 & & \\ & & \ddots & \\ & & & \boldsymbol{L}_k \end{bmatrix} \tag{12}$$

$\boldsymbol{L}$ is a block diagonal matrix, so its eigenvalues and eigenvectors are the eigenvalues and eigenvectors of these blocks. Since $\{\boldsymbol{L}_i\}^k_{i=1}$ is the Laplace matrices of every connected subgraph, according to properties *3)* and *4)*, every block matrix has a eigenvalue 0 and its corresponding eigenvector is the constant vector $\{\boldsymbol{l}_i\}^k_{i=1}$ in the connected subgraph. Therefore, the multiplicity of eigenvalue 0 of $\boldsymbol{L}$ is $k$, and the eigenvectors corresponding to eigenvalue 0 are $\boldsymbol{l}_{A_1}, \boldsymbol{l}_{A_2}, \cdots, \boldsymbol{l}_{A_k}$ .

Therefore, the following theorem can be obtained:

*Theorem* 1: Let $G$ be a undirected graph with nonnegative weights, in which $k$ connected subgraphs $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_k$ are used. The multiplicity of eigenvalue 0 of $\boldsymbol{L}$ is the same as that of connected subgraphs $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_k$, and the corresponding eigenvectors $\boldsymbol{l}_{A_1}, \boldsymbol{l}_{A_2}, \cdots, \boldsymbol{l}_{A_k}$ divides $k$ eigenspaces.

- Proposed spectral clustering

The proposed spectral clustering is based on the irregular Laplace matrix. If the matrix $\boldsymbol{F} \in \boldsymbol{R}^n$ contains $k$ orthogonal vectors $\boldsymbol{f}_1, \boldsymbol{f}_2, \cdots, \boldsymbol{f}_k \in \boldsymbol{R}^n$ , the objective function of the irregular spectral clustering is:

$$\begin{aligned} \min_{F} \quad & tr\left(\boldsymbol{F}^{\mathrm{T}} \boldsymbol{L} \boldsymbol{F}\right) \\ s.t. \quad & \boldsymbol{F}^{\mathrm{T}} \boldsymbol{F}=\boldsymbol{I} \end{aligned} . \tag{13}$$

In function (13), tr( • ) is the trace of the matrix. The solution of the above formula consists of $k$ eigenvectors corresponding to the $k$ minimum eigenvalues of $\boldsymbol{L}$. $\boldsymbol{F}$ can be regarded as the mapping of raw data in low dimensional space. After that, the traditional clustering algorithm, such as $k$-means, can be used to cluster $\boldsymbol{F}$ to get the cluster label of each point (user). The specific algorithm flow is shown in Table II.

The proposed algorithm can be implemented in a stable time and rerun in successive stable time segment.

Due to the calculation of eigenvalues and eigenvectors of a matrix in spectral clustering, the time and space complexity are high. With increased data, the time and space of running spectral clustering will increase dramatically. Spectral clustering only needs adjacent matrix, so it is very effective for clustering sparse data. Spectral clustering is very sensitive to the change of weight graph and the selection of clustering parameters. Compared with the traditional clustering algorithm, it can cluster the sample space of any shape and converge to the global optimal solution. If constraint information is added to spectral clustering, clustering effect can be improved.

TABLE II. SECTRAL CLUSTERING FLOW

| |
|---|
| **INPUT: The interference matrix *W*, Degree matrix *D*, user set *U*, Parameter *k*** |
| **OUTPUT: users clusters $V_i$, $i$ = 1, …, *n*** |
| 1. **Calculate** Laplace matrix $L = D - W$; |
| 2. **Calculate** the eigenvalues of $L$, **Sort** the eigenvalues from small to large, **take** the first *k* eigenvalues, and **calculate** the *k* eigenvectors $l_1, l_2, \cdots, l_k$; |
| 3. **Make** the above *k* column vectors into a matrix $L' = \{l_1, l_2, \cdots, l_k\}, L' \in R^{n \times k}$; |
| 4. **Let** $z_i \in R^k$ be the vector of line *i* of $L'$, where $i = 1, \dots, n$; |
| 5. **Use** *k*-means algorithm to reset the sample point $Z = \{z_1, z_2, \cdots, z_n\}$ to get cluster $V_1, V_2, \dots, V_k$. |

## IV. SIMULATION AND ANALYZATION

A group of users under several APs are clustered in the simulation. Users are scheduled among cells randomly in order to simulate the randomness of user locations. Clustered users are selected by greedy scheduling algorithm, in consideration of proportional fairness and rate maximization. For the APs in virtual cells, the power is allocated by the intensity of the instantaneous channel of the scheduled user. Power is controlled through water-filling algorithm.

### *A. Simulation Conditions*

In clustering process, the merging criteria is to minimize the sum of weights intra cluster and maximize weights inter cluster. Simulation parameters are shown in Table III.

TABLE III. SIMULATION PARAMETERS

| Parameters | |
|---|---|
| Cell radius | 100m |
| Frequency reuse factor | 1 |
| number of channels | 2 |
| noise power / (dBm) | *N*=-173.9+10*log10(10.^7)+9 |
| propagating power of a Pico base station / (dBm) | 20 |
| number of transmitting antennas | 36 |
| number of receiving antennas | 6 |

For fairness and authenticity of the simulation results, parameter setting, power partition and power control algorithms are identical in the compared algorithm and the proposed algorithm.

### *B. Results and Analysis*

A dense Network consisting 6 APs is simulated. Moreover, each AP are surrounded with 6 random-distributed users, which are as far away from APs as possible. So we can observe the improvement of the proposed algorithm about cell-edge users. Fig.5 shows the geographical distribution of random users and fixed BSs.

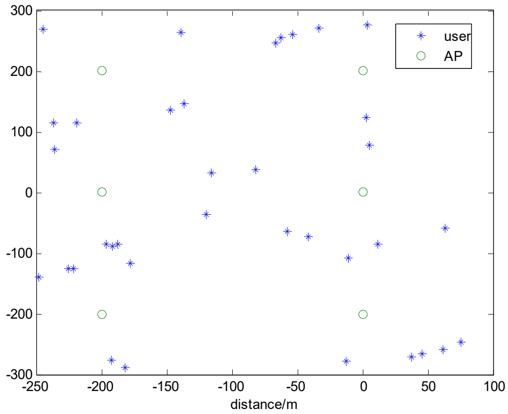


Fig. 5. Distribution of uses and APs

We take the parameter *k* as 2 as an example in simulation. The SE cdf curves of edge UE devices for each considered scheme are given in Fig. 6. It can be observed that the our proposed CoMP scheme achieve higher SE than the other two schemes, LBG[10] and *k*-means. Targeting the 10th percentile of the CDF as QoS measure, 16.91% and 5.37% improvement is observable of the proposed algorithm over *k*-means and LBG algorithm respectively. Targeting the 85th percentile, the improvement is also promising (6.19% and 5.49% over *k*-means and LBG, respectively).

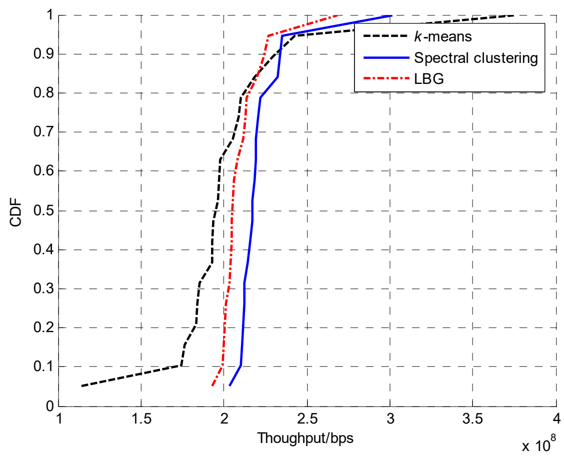


Fig. 6. CDF of different algorithms

Fig.7 shows the average user throughput of different clustering schemes. It shows that the proposed spectral clustering scheme can achieve higher throughput than the other two algorithms ——*k*_means and LBG. Because we set the users on the edge of a cell, Fig. 7 also shows the improvement of the cell-edge users' performance.

In dense small-cell networks, the complexity influences the system performance. Hence, we compare the running time of three schemes in Fig. 8 to evaluate the algorithm complexity. It can be seen from Fig. 8 that our proposed algorithm becomes the lowest of the three schemes. The spectral clustering scheme achieve higher throughput with relatively lower complexity.

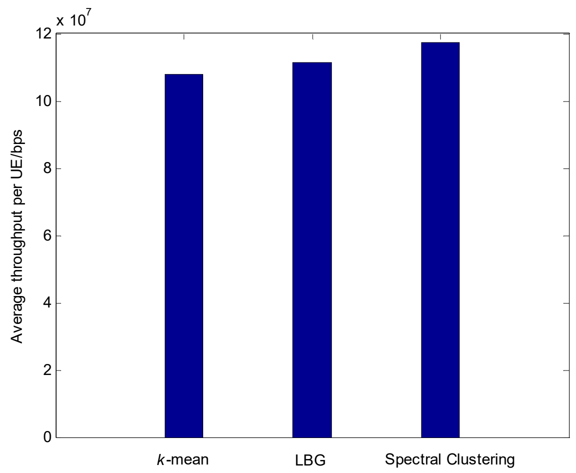


Fig. 7. Average user throughtput of different clustering schmenes ,where throughput is calculated using (3).

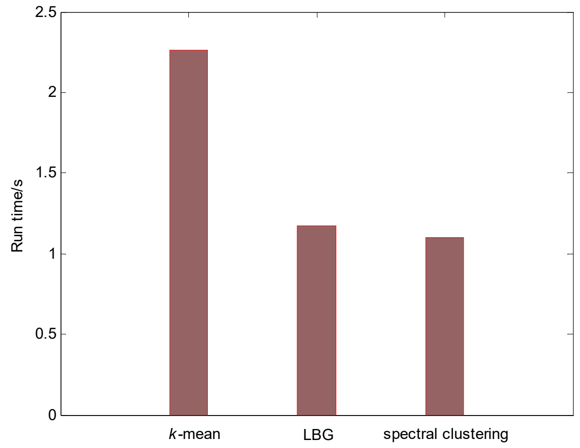


Fig. 8. Running time comparison

## V. CONCLUSION

In this paper, we have proposed a dynamic user clustering scheme for UDN to improve system and cell-edge users' throughput with consideration of complexity. Our scheme mainly consists of weight design, graph construction and spectral clustering. The CoMP users are distributed randomly and we use the proposed algorithm to cluster them into different clusters with the criteria of smallest interference inside a cluster. The performance of the proposed scheme is evaluated comparing with some existing schemes in the simulation. The results show that our scheme can increase user SE and system throughput particularly for cell-edge users. Moreover, our scheme requires reduced complexity as compared with other clustering algorithms, proving to be more practical in UDN.

## REFERENCES


[1] Chung Byung Chang, Cho Dong-Ho, "Semidynamic Cell-Clustering Algorithm Based on Reinforcement Learning in Cooperative Transmission System," IEEE Systems Journal, Vol. 12 No.4, 2018, pp.3853-3856.

[2] Lin Yan , Zhang Rong , Yang Luxi , Hanzo Lajos, "Secure User-Centric Clustering for Energy Efficient Ultra Dense Networks: Design and Optimization," IEEE Journal on Selected Areas in Communications, Vol. 36, No. 7, 2018, pp.1609-1621.

[3] David Lopezperez, Ming Ding, Holger Claussen, Amir H. Jafari, "Towards 1 Gbps/UE in Cellular Systems: Understanding Ultra-Dense Small Cell Deployments," IEEE Communications Surveys & Tutorials, Vol. 17, No. 4, 2015, pp.2078-2101.

[4] Ge Xiaohu, Tu Song, Mao Guoqiang, Wang Chengxiang, "5G Ultra-Dense Cellular Networks," IEEE Wireless Communications, Vol.23, No.1, 2016, pp. 72-79. DOI: 10.1109/MWC.2016.7422408

[5] Yang Bin, Mao Guoqiang, Ding Ming, Ge Xiaohu, Tao Xiaofeng, "Dense Small Cell Networks: From Noise-Limited to Dense Interference-Limited," IEEE Transactions on Vehicular Technology, Vol. 67, No. 5 ,2018, pp.4262-4277.DOI: 10.1109/ TVT.2018.2794452

[6] Z. Zhang, Ning Wang , Zhang Jiankang ,Mu Xiaomin, "Dynamic User-Centric Clustering for Uplink Cooperation in Multi-Cell Wireless Networks," IEEE Access, Vol. 6, 2018, pp.8526-8538.

[7] H. Cao, S. Wu, Y. Hu, R. Mann, Y. Liu, L. Yang, and H. Zhu, "An efficient energy cost and mapping revenue strategy for inter-domain NFV-enabled networks," IEEE Internet of Things Journal, vol. 7, no. 7, pp. 5723-5736, Jul. 2020.

[8] D. Gesbert, S. Hanly, H. Huang, S. Shamai (Shitz), O. Simeone, and W. Yu, "Multi-cell MIMO cooperative networks: A new look at interference," IEEE J. Sel. Areas Commun., vol. 28, no. 9, pp. 1380_1408, Dec. 2011.

[9] P. Marsch , G. Fettweis, "Static Clustering for Cooperative Multi-Point (CoMP) in Mobile Communications," IEEE International Conference on Communications, Jun 5-9, 2011, Kyoto, Japan.

[10] S.Nam , J. Oh and Y. Han, " A dynamic transmission mode selection scheme for CoMP systems," IEEE International Symposium on Personal Indoor & Mobile Radio Communications, Aug. 31 – Sep. 3, 2020, London, UK.

[11] X. H. You, D. M. Wang, B. Sheng, et al., "Cooperative distributed antenna systems for mobile communications [Coordinated and Distributed MIMO," IEEE Wireless Communications, vol. 17, no.3, 2010, pp.35-43.

[12] A. Papadogiannis, D. Gesbert and E. Hardouin, "A Dynamic Clustering Approach in Wireless Networks with Multi-Cell Cooperative Processing," IEEE International Conference on Communications(ICC) 2008, pp. 4033-4037. ICC 2008, 19-23 May 2008, Beijing, China.

[13] Haijun Zhang, Hui Liu, Chunxiao Jiang, Xiaoli Chu, A. Nallanathan and Xiangming Wen, " A Practical Semidynamic Clustering Scheme Using Affinity Propagation in Cooperative Picocells," IEEE Trans. On Vehicular Technology, Vol.64, No. 9, 2015, pp.4372-4377.

[14] Siqiang Ke, Yujie Li, Zhibin Gao, Lianfen Huang, "An Adaptive Clustering Approach for Small Cell in Ultra-Dense Networks," 9th International Conference on Advanced Infocomm Technology, 22-24 November 2017, Chengdu, China.

[15] Niu C, Li Y, Hu R Q, et al, "Fast and Efficient Radio Resource Allocation in Dynamic Ultra-Dense Heterogeneous Networks," IEEE Access, Vol. 5, No. 99, 2017, pp.1911-1924.

[16] Yanxia Liang, Yao Liu, Changyin Sun, Xin Liu, Jing Jiang, Yan Gao, Yongbing Xie, "An Improved LBG Algorithm for User Clustering in Ultra-Dense Network," Proceedings of the Fifth Euro-China Conference on Intelligent Data Analysis and Applications, October 2018, pp.44-52.

[17] Grevan M., Newman M. E. J., "Community structure in social and biological networks," Proceedings of National Academy of Science, Vol. 99, No. 12, 2002, pp. 7821-7826.

[18] Xianchao Zhang, Data Clustering, Bejing: Science Press, 2017, pp. 157-281,and 261-281.

[19] H. Cao, S. Wu, Y. Guo, H. Zhu, and L. Yang, "Mapping strategy for virtual networks in one stage," IET Communications, vol. 13, no. 14, pp. 2207-2215, 2019.

[20] H. Cao, H. Zhu, and L. Yang, "Collaborative attributes and resources for single-stage virtual network mapping in network virtualization," Journal of Communications and Networks, vol. 22, no. 1, pp.61-71, Feb. 2020.

[21] Y. Liang, C. Sun, J. Jiang, X. Liu, H. He and Y. Xie, "An Efficiency-Improved Clustering Algorithm Based on KNN Under Ultra-Dense Network," IEEE Access, vol. 8, 2020, pp. 43796-43805.
doi: 10.1109/ACCESS.2020.2977403.

[22] W. E. Donath, A. J. Hoffman, "Lower bounds for the partitioning of graphs," IBM Journal of Research & Development, Vol. 15, No.5, 1973, pp.420-425.