QoS-Based Interference Alignment with Similarity Clustering for Efficient Subchannel Allocation in Dense Small Cell Networks

Hao Zhang, Hongyan Li, Member, IEEE, Jung Hoon Lee, Member, IEEE, and Huaiyu Dai, Fellow, IEEE

Abstract—Interference alignment (IA) can remarkably improve the spectral efficiency of dense small cell networks (SCNs) underlaying a macrocell, but its feasibility condition and implementation complexity are restricted by the number of SUEs. Moreover, the SUEs performing IA may have unsatisfactory quality of service (QoS) requirements as IA only eliminates interference while neglecting the gain of desired signals. In this paper, we propose a centralized efficient subchannel allocation scheme based on IA with similarity clustering in dense SCNs underlaying a macrocell, which aims at maximizing the number of QoS guaranteed SUEs performing IA. The corresponding problem is formulated as a combinatorial optimization problem which is NP-hard. So a low-complexity solution is proposed which includes three phases: similarity clustering for SUEs through graph partitioning, further adjustments of cluster sizes to make IA feasible in each cluster and subchannel allocation for the formed clusters, each of which is performed with a notably reduced computational complexity. Moreover, the proposed solution greatly reduces the signaling overhead incurred by channel state information (CSI) estimation. Numerical results show that the proposed solution not only outperforms other related schemes, but also achieves a performance close to the optimal solution.

Index Terms—Dense small cell networks, interference alignment, QoS guarantee, similarity clustering, subchannel allocation.

I. INTRODUCTION

The dramatic growth in the number of users and their demands for mobile data traffic require higher network capacity as well as enhanced quality-of-service (QoS) requirements in 5G wireless communication systems. Such requirements become demanding especially in indoor areas [1] such as homes and offices, where the users experience poor coverage. One promising solution is to densely deploy low-power and low-cost small cell base stations (SBSs) in indoor areas [2], [3], and allow the small cell equipments (SUEs) to reuse the limited spectral resources already allocated to the macrocell user equipments (MUEs), which also provides a higher spectral efficiency and will be attractive to the mobile operators.

Mr. Hao Zhang and Prof. Hongyan Li are with the State Key Laboratory of Integrated Service Networks, Xidian University, Xi’an, Shaanxi, 710071, China (e-mail: zhanghaoxdisn@126.com; hyli@xidian.edu.cn).

Prof. Jung Hoon Lee is with the Department of Electronics Engineering, Hankuk University of Foreign Studies, Yongin 17055, South Korea (e-mail: tanheta@hufs.ac.kr).

Prof. Huaiyu Dai is with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, 27695, USA (e-mail: hdai@ncsu.edu).

However, in orthogonal frequency division multiple access (OFDMA) based dense small cell networks (SCNs) underlaying a macrocell, the co-channel deployment incurs both co-tier and cross-tier interference, which significantly degrade the network performance [4]. Thus, how to efficiently manage aforementioned two types of interference is one of the most critical issues [1], [5]. Also, due to the dramatic growth in the number of users, it is almost impossible to mitigate the additional interference only with the traditional interference management techniques such as subchannel allocation and power control [6]. So it is of great importance to exploit advanced interference management techniques for interference mitigation.

Recently, as a promising technique of interference management, interference alignment (IA) has attracted much attention. Its basic concept is to consolidate multiple interference signals into a reduced-dimensional subspace, while the remaining subspace is used to recover the desired signals free from interference at each receiver [7]. In [8], it was shown that for K-user interference channel with M (M > 1) antennas at each transmitter and receiver, the maximum degrees of freedom (DoF) achieved by IA in high signal-to-noise ratio (SNR) regime is KM/2. Therefore, it is expected that IA can help to improve the spectral efficiency of wireless networks especially where the limited spectrum resources are inadequate to support all active users.

However, the application of IA is restricted by its feasibility condition [9] and implementation complexity due to the design of precoding and decoding matrices [10]. Clustered IA [11], [12] is proposed to ameliorate the aforementioned two issues. Its basic concept is to group all users into disjoint clusters, each of which contains a limited number of users meeting the feasibility condition of IA. In this way, the implementation complexity is also reduced. However, IA only focuses on eliminating the interference but pays no attention to the gain of desired signals at each receiver [13], which may lead to a failure of maintaining the minimum data rate requirements (i.e., QoS requirements) after performing IA in low and moderate SNR regimes. And this issue is still not fully addressed by clustered IA. So, in dense SCNs, it is expected that resource allocation based on clustered IA is performed to guarantee SUEs’ enhanced QoS requirements.

In this paper, to mitigate both co-tier and cross-tier interference in dense SCNs underlaying a macrocell, a centralized interference management scheme of QoS-based IA...
with similarity clustering for efficient subchannel allocation is proposed. The corresponding problem is formulated as a combinatorial optimization problem, which is generally known as NP-hard. Using exhaustive search to solve it will pose unaffordable burden on a central entity called Home eNB Gateway (HeNB GW) [14] in a practical system, and will also require to estimate the accurate global channel state information (CSI) which results in heavy signaling overhead. So we propose to solve it through three phases. In the first phase, the HeNB GW groups all SUEs into disjoint clusters by partitioning the constructed similarity graph, this is because the optimal clustering for SUEs can only be obtained by exhaustive search, which will incur huge computational burden on HeNB GW. In the second phase, the HeNB GW makes further adjustments of clusters sizes to make IA feasible in each cluster which is not able to be guaranteed in the first phase. As performing the aforementioned two phases only needs the information of path losses and each SUE's QoS requirement instead of the accurate global CSI, the signaling overhead is greatly reduced. In the third phase, constraints of inter-cluster and cross-tier interference are further relaxed to make the accurate CSI estimation be confined in each of the formed cluster, which significantly reduces the signaling overhead; then greedy-style efficient algorithms are proposed to solve the subproblem of subchannel allocation for clusters. Each phase can be performed by the HeNB GW with low-complexity. Finally, the co-tier interference, including intra-cluster and inter-cluster interference, is efficiently mitigated, i.e., the intra-cluster interference is completely eliminated by IA with similarity clustering, while the inter-cluster interference is mitigated by subchannel allocation; and the cross-tier interference incurred by SBSs at each MUE is also mitigated by subchannel allocation. Our main contributions are summarized as follows.

- We propose an efficient subchannel allocation scheme based on IA with similarity clustering to mitigate both co-tier and cross-tier interference in dense SCNs underlying a macrocell, which is centralized and maximizes the number of QoS guaranteed SUEs performing IA.
- Due to the NP-hardness of corresponding optimization problem, we propose a three-phases efficient solution with low-complexity, which provides a performance close to the optimal solution and greatly reduces the signaling overhead of CSI estimation.
- We group all SUEs into disjoint clusters according to similarities in both QoS requirements and path losses through partitioning the similarity graph in the first phase, which notably reduces the complexity of clustering and signaling overhead of CSI estimation.
- We make further adjustments of cluster sizes to ensure IA is feasible in each cluster by the proposed algorithm in the second phase, which cannot be guaranteed in the first phase.
- We propose greedy-style efficient algorithms for HeNB GW to allocate subchannels to the formed clusters, which has notably reduced computational complexity and signaling overhead of CSI estimation.
- We analyze the computational complexities of the optimal solution and the proposed solution to demonstrate that the proposed solution has a notably reduced computational complexity.

The rest of this paper is organized as follows. The related work is reviewed in Section II. The system model and problem formulation are presented in Section III. A three-phases efficient solution to the formulated problem is proposed in Section IV. The computational complexities of proposed solution as well as the optimal solution are analyzed in Section V. The numerical results are shown in Section VI. Finally, the paper is concluded in Section VII.

**Notations:** The bold lower case and upper case letters represent vectors and matrices, respectively. $\mathbb{R}^{N_R \times N_T}$ and $\mathbb{C}^{N_R \times N_T}$ are the sets of real and complex $N_R \times N_T$ matrices, respectively. $N(a, A)$ and $CN(a, A)$ denote the real and complex Gaussian distributions with mean $a$ and covariance matrix $A$, respectively. $I_{d_0}$ and $\Omega_{d_0}$ are the $d_0 \times d_0$ identity matrix and $d_0 \times d_0$ matrix with elements being all 0, respectively. $|C|$ denotes the cardinality of $C$. $\|a\|_1$ and $\|a\|_2$ represent the $l_1$-norm and $l_2$-norm of vector $a$, respectively. $E\{\cdot\}$ is the operation of expectation. $A^H$, $\text{rank}(A)$, $A^{-1}$, $|A|$ and $\text{tr}(A)$ denote the Hermitian transpose, rank, inverse, determinant and trace of matrix $A$, respectively. Finally, $\cup$, $\cap$ and $\setminus$ are the union, intersection and subtraction operations of sets, respectively.

**II. RELATED WORK**

Several works have investigated the problem of clustered IA. In [11] and [12], clustered IA was first proposed for cellular and ad hoc networks, respectively. However, how to form disjoint clusters by specific criteria is not investigated. To the best of our knowledge, clustering for users performing IA can be divided into two categories in existing literatures. The first one is clustering by path losses or distances [15]–[17]. In [15], a criterion based on path losses was proposed to group all the transmitter-receiver (Tx-Rx) pairs performing IA into disjoint clusters. With this criterion, strong interference was captured as intra-cluster interference and completely eliminated by clustered IA, while the relatively weak interference was left as inter-cluster interference and treated as noise. However, in dense SCNs, the inter-cluster interference cannot be simply treated as noise any more when the SBSs in one cluster are closer to SUEs in other clusters. In [16], another criterion based on path losses was used to select part of the Tx-Rx pairs to form a single cluster performing IA, which ensured the strength of interference would be close to those of desired signals in the formed cluster. Then interference detection was exploited at each receiver in the formed cluster to remove the residual interference induced by the transmitters not performing IA. But forming only one cluster is far from enough for interference mitigation in dense SCNs. In [17], the whole network was divided into an IA subnetwork (i.e., a cluster) and several spatial multiplexing (SM) subnetworks according to distances, where each SM subnetwork consisted of only one user. However, there are no resource allocation process based on clustered IA in [15]–[17], and the QoS requirement at each receiver after performing IA is not considered, either.
The second category, which also performs resource allocation based on IA with clustering for interference mitigation, is clustering directly by achievable data rate [18]–[21]. In essence, this category performs clustering and resource allocation for clusters performing IA simultaneously. In [18], one combination of Tx-Rx pairs (i.e., a cluster) that could achieve maximum data rate was selected to perform IA over each transmission time interval in femtocell networks. But the clustering process needs to exhaust all the possible combinations, which may incur prohibitive computational complexity when the number of Tx-Rx pairs becomes larger. In [19], the authors utilized coalitional game in the partition form to group femtocells into disjoint coalitions performing IA to maximize the achievable data rate of each femtocell base station. However, the subcarriers allocated to femtocell users were orthogonal to those allocated to MUEs, which will reduce the spectral efficiency. In [20], the phase of frequency-clustering allocated each subcarrier to a cluster that could achieve the maximum rate over it after performing IA, whose size met the feasibility condition of IA. However, how to guarantee the users’ QoS requirements after performing IA in each cluster is still not investigated in [18]–[20]. In [21], the authors extended their work in [20] by considering each secondary user’s (SU’s) QoS requirement after performing IA, where the SU’s QoS constraints were relaxed by minimizing the number of SUs whose rates were below the given threshold. However, all the formed clusters have equal size, which is not a general case of clustering. Also, each SU transmitter is assumed to cause strong interference to all other SU receivers, which is impractical in dense SCNs. The number of all possible clustering results grows exponentially with the number of SUEs in dense SCNs, so the second category will incur unaffordable computational complexity and should be avoided.

It is also noteworthy that a variety of applications call for differentiated QoS requirements for SUEs [22], [23]. Therefore, a higher spectral efficiency can be achieved if the SUES with similar QoS requirements are grouped into the same clusters, as they need almost the same number of subchannels to meet their QoS requirements. Meanwhile, the path losses from SBSs to the SUEs they do not serve in each cluster should also be similar, since it is unnecessary to eliminate weak interference induced by large path losses through clustered IA. However, when grouping users into disjoint clusters, none of the aforementioned works consider either the similarity in users’ QoS requirements or the similarities in both QoS requirements and path losses.

III. System Model And Problem Formulation

A. System Model

As depicted in Fig. 1, we consider the downlink transmission of OFDMA-based SCNs overlaying a single macrocell where all the SUEs share the same subchannel resources with MUEs [24], [25]. All SUEs served by SBSs are densely deployed in deep indoor area where all SUEs are out of the coverage of MBS [22]. Therefore, the cross-tier interference induced by the macrocell base station (MBS) at each SUE is considered as circularly symmetric additive white Gaussian noise (AWGN). All MUEs served by MBS are deployed in outdoor area, and the SBSs will cause cross-tier interference to the outdoor MUEs near them. Furthermore, the subchannel allocation for MUEs has been finished by MBS in a previous stage, since the MUEs are given higher priority to utilizing subchannel resources when the dense SCNs are underlaying a macrocell, which is also assumed in [1], [26]. Then MBS sends the results of subchannel allocation for MUEs to the HeNB GW.

We denote the set of small cells by $K = \{1, 2, \ldots, K\}$. In addition, we assume all SUEs have already been associated with their serving SBSs and this association also keeps fixed during the whole process of clustering and resource allocation [1], [19], [27]. Each SBS serves multiple SUEs. In this paper, for simplicity, we assume each SBS exclusively serves one SUE in a given time slot of subchannel allocation\(^1\) [27]. Then, in a new time slot, the HeNB GW will repeatedly perform the whole process of clustering and subchannel allocation. Here the exclusive SUE served by SBS $k$ in each given time slot is called SUE $k$. Let $\mathcal{N} = \{1, 2, \ldots, N\}$ denote the set of subchannels available in each time slot of subchannel allocation, where $N < K$. Each SBS has $N_T$ transmit antennas, and each SUE and MUE has $N_R$ receive antennas. Meanwhile, each SBS sends $d_0$ data streams to its served SUE, where $d_0 \leq \min\{N_T, N_R\}$ [9].

Let $C_i$ be an arbitrary cluster contained by $K$. Since co-tier interference suffered by SUE $k$ in cluster $C_i$ consists of both intra-cluster and inter-cluster interference, the received signal

\(^1\)Note that how to select an exclusive SUE for an SBS in a given time slot is beyond the scope of our paper, so this process is assumed to have been finished before clustering and subchannel allocation.
at SUE $k$ in $C_i$ over subchannel $n$ is expressed as

$$y_k^n = \sqrt{1/\mathbb{P}L_{kk}} H_{kk}^n V_k^n s_k^n + \sum_{j \in C_i \setminus \{k\}} \sqrt{1/\mathbb{P}L_{kj}} H_{kj}^n V_j^n s_j^n$$

\[ \text{ intra-cluster interference} \]

$$+ \sum_{l \in K \setminus C_i} \sqrt{1/\mathbb{P}L_{kl}} H_{kl}^n V_l^n s_l^n + n_k^n,$$

\[ \text{ inter-cluster interference} \]

where $\mathbb{P}L_{kj}$ is the path loss from SBS $j$ to SUE $k$. $H_{kj}^n \in \mathbb{C}^{N_T \times N_T}$ is the channel matrix between SBS $j$ and SUE $k$ over subchannel $n$. $V_k^n \in \mathbb{C}^{N_T \times d_0}$ denotes the transmit precoding matrix of SBS $k$ over subchannel $n$. $s_k^n \in \mathbb{R}^{d_0 \times 1}$ is the vector consisting of symbols transmitted from SBS $k$ to its served SUE $k$ such that $s_k^n \sim \mathcal{CN}(0_{N^T \times 1}, \mathbf{p}_k^n \mathbf{I}_{d_0})$, where $\mathbf{p}_k^n$ is the power of each independently encoded Gaussian codebook symbol in $s_k^n$, and $\mathbf{p}_k^n$ is the transmit power of SBS $k$ over subchannel $n$. Since each symbol is beamformed with the corresponding column of $V_k^n$, it should be satisfied that $E[\|V_k^n s_k^n\|^2] = \mathbf{p}_k^n$ [13]. As we only focus on the sub-channel allocation for SUEs, we assume the transmit power of each SBS over each subchannel is fixed to a constant $p$ during the whole process of clustering and subchannel allocation, i.e., $\mathbf{p}_k^n = p$, $\forall K$, $\forall n \in N$. $n_k^n \in \mathbb{C}^{N_R \times 1}$ is AWGN at SUE $k$ over subchannel $n$ such that $n_k^n \sim \mathcal{CN}(0_{N_R \times 1}, (\sigma_k^n)^2 I_{N_R})$, where $(\sigma_k^n)^2$ is the noise power at SUE $k$ over subchannel $n$.

According to the feasibility condition of IA in [9], it should satisfy that

$$d_0 \leq \frac{N_T + N_R}{|C_i| + 1}, \forall C_i \subseteq K. \quad (2)$$

So the size of cluster $C_i$ must meet the following inequality:

$$|C_i| \leq \left\lfloor \frac{N_T + N_R}{d_0} \right\rfloor \leq S_{\max}, \forall C_i \subseteq K. \quad (3)$$

Note that the larger the size of one cluster, the higher spectral efficiency can be achieved by IA in this cluster. So the number of clusters is determined by

$$\Gamma = \left\lfloor \frac{K}{S_{\max}} \right\rfloor + 1, \quad (4)$$

where $\left\lfloor \frac{K}{S_{\max}} \right\rfloor$ is the integer largest smaller than or equal to $\frac{K}{S_{\max}}$. To enable all SUEs in cluster $C_i$ to perform IA over subchannel $n$, the following two equations must also be met [13]:

$$\begin{cases} 
(U_k^n)^H H_{kk}^n V_j^n = 0_{d_0}, \\
\text{rank} \left(U_k^n H_{kk}^n V_k^n\right) = d_0, 
\end{cases} \quad \forall k \in C_i, \forall j \in C_i \setminus \{k\}, \quad (5)$$

where $U_k^n \in \mathbb{C}^{N_R \times d_0}$ is the interference suppression matrix at SUE $k$. Therefore, after performing IA, the received signal at SUE $k$ over subchannel $n$ becomes

$$\begin{aligned}
(U_k^n)^H y_k^n &= \sqrt{1/\mathbb{P}L_{kk}} (U_k^n)^H H_{kk}^n V_k^n s_k^n + \\
&\sum_{j \in C_i \setminus \{k\}} \sqrt{1/\mathbb{P}L_{kj}} (U_k^n)^H H_{kj}^n V_j^n s_j^n + (U_k^n)^H n_k^n,
\end{aligned}$$

in which the intra-cluster interference is perfectly eliminated by clustered IA, leaving only the inter-cluster interference as co-tier interference. Then the achievable rate of SUE $k$ after performing IA over subchannel $n$ is given by

$$R_k^n = \log \left| I_{d_0} + \frac{p}{d_0 \cdot \mathbb{P}L_{kk}} (U_k^n)^H H_{kk}^n V_k^n (V_k^n)^H (H_{kk}^n)^H U_k^n \times \\
\sum_{j \in K \setminus C_i} \frac{p}{d_0 \cdot \mathbb{P}L_{kl}} (U_k^n)^H H_{kl}^n V_l^n (V_l^n)^H (H_{kl}^n)^H U_k^n \right|^{-1}. \quad (7)$$

Furthermore, as the SUEs reuse the subchannels allocated to MUEs, the cross-tier interference induced by SBS $k$ at MUE $m$ over its allocated subchannel $n_{mk}$ can be evaluated as [21]²

$$I_{mk}^{n_{mk}} = \text{tr} \left( \Omega_m^{n_{mk}} \frac{p}{d_{0} \cdot \mathbb{P}L_{mk}} H_{mk}^{n_{mk}} V_k^{n_{mk}} (V_k^{n_{mk}})^H (H_{mk}^{n_{mk}})^H \right), \quad (8)$$

where $\Omega_m^{n_{mk}}$ is the interference factor of subchannel $n_{mk}$ to MUE $m$. $H_{mk}^{n_{mk}}$ is the channel matrix between SBS $k$ and MUE $m$ over subchannel $n_{mk}$. Similar to (8), the inter-cluster interference induced by SBS $k$ in $C_i$ at SUE $l$ in $K \setminus C_i$ over subchannel $n$, i.e., $I_{lk}^{n}$, can also be evaluated.

B. Problem Formulation

In dense SCNs underlaying a macrocell, the number of SUEs is much larger than that of subchannels available, so it may be unable to guarantee the QoS requirements for all SUEs performing IA. Aiming at maximizing the number of QoS guaranteed SUEs, the corresponding optimization problem of clustering and subchannel allocation for SUEs is formulated as

$$\max_{C_i, \rho_{k}^m} \|q\|_1 \quad (9)$$

²In [21], formula (8) was used to evaluate the interference induced by an SU transmitter to a primary user (PU) receiver over a subchannel that has already been allocated to this PU receiver. Here SBS $k$ and MUE $m$ can be considered as an SU transmitter and a PU receiver, respectively.
s.t. C1: \( \mathcal{C}_i \cap \mathcal{C}_s = \emptyset, \ \forall \mathcal{C}_i, \mathcal{C}_s \subseteq \mathcal{K} \)

C2: \( \bigcup_{i=1}^{\Gamma} \mathcal{C}_i = \mathcal{K}, \ \forall \mathcal{C}_i \)

C3: \( |\mathcal{C}_i| \leq |\mathcal{S}|_{\text{max}}, \ \forall \mathcal{C}_i \)

C4: \( \sum_{n \in \mathcal{N}} \rho_{i0}^m \geq 1, \ \forall \mathcal{C}_i \)

C5: \( \rho_{i0}^m \sum_{k \in \mathcal{C}_i} I_{nk}^m \leq I_{nk}^m, \ \forall m, \ \forall \mathcal{C}_i, \ \forall n \in \mathcal{N}_m \subseteq \mathcal{N} \)

C6: \( \rho_{i0}^m \sum_{k \in \mathcal{C}_i} I_{nk}^m \leq I_{nk}^m, \ \forall n, \ \forall \mathcal{C}_i, \ \forall l \in \mathcal{K} \setminus \{\mathcal{C}_i\} \)

C7: \( q_k = \begin{cases} 1, & \text{if } \sum_{n \in \mathcal{K}} \rho_{i0}^m R_{nk}^m \geq R_k, \ \forall k \in \mathcal{C}_i, \ \forall \mathcal{C}_i, \\ 0, & \text{otherwise} \end{cases} \)

C8: \( \rho_{i0}^m \in \{0, 1\}, \ \forall \mathcal{C}_i, \ \forall n. \)

In problem (9), \( \|q\|_1 \) is the \( l_1 \)-norm of indicator vector \( q \), where \( q \triangleq [q_1, \ldots, q_K] \), and \( q_k \) is a binary variable which equals \( 1 \) if SUE \( k \) achieves satisfactory QoS requirement after performing IA over its allocated subchannels and \( 0 \) otherwise. So \( \|q\|_1 \) is also the number of QoS guaranteed SUEs. C1 and C2 indicate that all the \( \Gamma \) clusters are disjoint and constitute the entire set of SUEs \( \mathcal{K} \). C3 limits the maximum size of each cluster performing IA. In C4, \( \rho_{i0}^m \) is a binary variable, which equals \( 1 \) if subchannel \( n \) is allocated to cluster \( C_i \) and \( 0 \) otherwise. C4 ensures that each cluster is allocated with at least one subchannel, which is for the fairness among disjoint clusters. C5 requires that the total cross-tier interference induced by SBSs in an arbitrary cluster \( C_i \) over subchannel \( n \) in \( \mathcal{N}_m \) at MUE \( m \) should not exceed the threshold \( I_{nk}^m \) which is prescribed by MBS. Here \( \mathcal{N}_m \) is a known set consisting of subchannels allocated to MUE \( m \) such that \( \bigcup_m \mathcal{N}_m = \mathcal{N} \). C6 ensures that the total interference induced by SBSs in \( C_i \) at SUE \( l \) in \( \mathcal{K} \setminus \{C_i\} \) over subchannel \( n \) should not exceed threshold \( I_{nk}^m \) which is prescribed by the HeNB GW.

IV. PROPOSED SOLUTION

Problem (9) is a combinatorial optimization problem which is generally known as NP-hard. Furthermore, constraints C5 and C6 require to estimate the accurate global CSI, i.e., the channel matrices between each SBS and all MUEs over each of their allocated subchannels and those between each SBS and all SUEs over each subchannel, which will incur huge signaling overhead. Therefore, to reduce the computational burden on HeNB GW as well as the signaling overhead of CSI estimation, we propose to solve it through three phases: 1) clustering based on similarities in both QoS requirements and path losses (CSQPL); 2) further adjustment of cluster sizes; and 3) efficient subchannel allocation for the formed clusters, each of which can be performed by HeNB GW with low-complexity. The similar method, i.e., clustering for users and then resource allocation for clusters, is also used in [1], [27] and [28].

\footnote{It is noteworthy that performing IA in cluster \( C_i \) enables all the SUEs in \( C_i \) to share subchannel \( n \) without intra-cluster interference. So allocating subchannel \( n \) to \( C_i \) is equivalent to allocating subchannel \( n \) to each SUE in \( C_i \).}

A. Phase 1: Clustering for SUEs Based on Similarities in QoS Requirements and Path Losses

The SUEs have differentiated QoS requirements. Besides, as the SCN nodes are randomly deployed in indoor area, the path losses from SBSs to the SUEs they do not serve are of great difference. To maximize the number of QoS guaranteed SUEs, it is expected that the clustering is based on similarities in both QoS requirements and path losses, which is due to the following two facts.

Firstly, in a cluster, the path losses from different SBSs to the SUEs they do not serve are similar, but the SUEs’ QoS requirements are dissimilar. In this case, the SUEs with higher QoS requirements should be allocated with more subchannels to meet their QoS requirements, while those with lower QoS requirements only need less subchannels. However, allocating one subchannel to a cluster performing IA means allocating this subchannel to each SUE in this cluster. Following this fact, we know that each SUE in this cluster is allocated with equal number of subchannels. Therefore, when we aim at maximizing the number of QoS guaranteed SUEs, more subchannels are only needed by SUEs with higher QoS requirements but quite redundant for the SUEs with lower QoS requirements since they have already had satisfactory QoS requirements after being allocated with less subchannels. In this way, the spectral efficiency will also be reduced. Furthermore, in a cluster, all SUEs have similar QoS requirements, but the path losses from SBSs to the SUEs they do not serve are dissimilar. In this case, the strength of interference at each SUE will be of great difference, so it is unnecessary for SUEs to eliminate the weak interference induced by SBSs with larger path losses to them through IA in this cluster.

Next, the HeNB GW uses the normalized spectral clustering algorithm [29] for similarity clustering, which leads to a notably reduced computational complexity. This algorithm first needs to construct the weighted similarity graph \( \mathcal{G} \triangleq (\mathcal{V}, \mathcal{E}) \), where \( \mathcal{V} \) and \( \mathcal{E} \) denote the sets of vertices and edges, respectively. Each SBS and its served SUE (i.e., each SBS-SUE pair) are represented as a vertex in \( \mathcal{V} \) such that \( |\mathcal{V}| = |\mathcal{K}| \). Let \( \mathcal{W} \triangleq \{w_{kj}|w_{kj} \geq 0, k, j \in \mathcal{V}\} \) be the weighted adjacency matrix of \( \mathcal{G} \), where \( w_{kj} \) is the weight of edge connecting vertices \( k \) and \( j \), which also measures the similarity between vertices \( k \) and \( j \). Since each cluster is expected to contain vertices with both similar QoS requirements and path losses, \( w_{kj} \) should be the decreasing function of both \( |R_k - R_j| \) and \( PL_{kj} \), where \( PL_{kj} \) is the path loss from vertex \( k \) to vertex \( v_k \). Then \( w_{kj} \) is defined as

\[
w_{kj} = \begin{cases} \exp \left( -\frac{|R_k - R_j|}{R_{\text{max}} - R_{\text{min}}} \right) \cdot \frac{PL_{kj}}{PL_{\text{max}}}, & \text{if } PL_{kj} \leq PL_0 \\ 0, & \text{otherwise} \end{cases}
\]

where \( \exp (\cdot) \) is the exponential function whose base is Euler’s
number $e$. Moreover, $R_{\text{max}}$ and $R_{\text{min}}$ are the maximum and minimum QoS requirements for SUEs in $\mathcal{K}$, respectively, i.e., $R_{\text{max}} = \max \left\{ R_k | k \in V \right\}$, and $R_{\text{min}} = \min \left\{ R_k | k \in V \right\}$. Since the similarity graph $\mathcal{G}$ is undirected, it is required that $w_{kj} = w_{jk}$, $\forall k, j \in V$. To guarantee this, $PL'_{kj}$ is defined as

$$PL'_{kj} = \frac{PL_{kj} + PL_{jk}}{2}, \quad \forall k, j \in V.$$ (11)

$PL'_{\text{max}}$ is the maximum path loss between any two vertices in $V$, i.e., $PL'_{\text{max}} = \max \left\{ PL_{kj} | k, j \in V \right\}$. It is also noteworthy that both $|R_k - R_j|$ and $PL'_{kj}$ are normalized in (10) because directly using the term $|R_k - R_j| \cdot PL'_{kj}$ will be meaningless. $PL_0$ is a prescribed threshold that simply makes a distinction between strong and weak interference, i.e., if the path loss between two vertices is smaller than $PL_0$, the interference between them will be relatively strong, otherwise it will be relatively weak and the weight of edge connecting these two vertices is 0.

Let $d_k$ be the total connection from vertex $k$ to all other vertices in $V$, which is defined as

$$d_k = \sum_{j \in V, j \neq k} w_{kj}. \quad (12)$$

Besides, let $d_{kj}$ be the element in $k$th row and $j$th column of a $K \times K$ diagonal matrix $D$, i.e.,

$$d_{kj} = \begin{cases} d_k, & \text{if } k = j, \forall k, j \in V. \\ 0, & \text{else} \end{cases} \quad (13)$$

Then the normalized Laplacian matrix of graph $\mathcal{G}$ is defined as [29]

$$L_{\text{norm}} = D^{-\frac{1}{2}} (D - W) D^{-\frac{1}{2}}. \quad (14)$$

Then the normalized spectral clustering algorithm is shown in Algorithm 1. Moreover, the path losses among vertices in each cluster do not exceed $PL_0$ after Algorithm 1.

To execute Algorithm 1, the HeNB GW only needs to know each SUE’s QoS requirement and path loss from each SBS to each SUE. So the estimation of accurate global CSI is avoided, and the signaling overhead is notably reduced. Each SUE can estimate the path losses from different SBSs to it by computing the ratios of the transmit power of pilot symbols from different SBSs to the power of corresponding received pilot symbols at this SUE. After finishing the estimation, each SUE reports the estimated path losses and its own QoS requirement to its serving SBS. Then, each SBS forwards these reported information to HeNB GW through S1 interface (wired backhaul) [14]. However, Algorithm 1 only partitions the similarity graph into $\Gamma$ clusters while paying no attention to constraint $C3$ in problem (9). So there may exist some clusters whose sizes exceed $S_{\text{max}}$ after Algorithm 1, which makes IA infeasible in these clusters. This is exactly what we try to avoid when exploiting IA.

Algorithm 1 Normalized Spectral Clustering Algorithm

1. Construct the similarity graph $\mathcal{G} = (V, E)$ and compute the weighted adjacency matrix $W$ according to (10).
2. Compute the diagonal matrix $D$ and the normalized Laplacian matrix $L_{\text{norm}}$ through (10) to (14).
3. Compute the eigenvectors $x_1, \ldots, x_{\Gamma}$ corresponding to the $\Gamma$ smallest eigenvalues of $L_{\text{norm}}$, and set $X = [x_1, \ldots, x_{\Gamma}]$.
4. Form the $K \times \Gamma$ matrix $X$ by normalizing each row of $X$, i.e., set $\tilde{x}_{kj} = x_{kj} / \left( \sum_{q=1}^{\Gamma} x_{kq}^2 \right)^{\frac{1}{2}}, \forall k \in V, \forall q \in \{1, 2, \ldots, \Gamma\}$.
5. Operate $k$-means algorithm algorithm to group the $K$ rows of $X$ into $\Gamma$ clusters.

B. Phase 2: Further Adjustments of Cluster Sizes

The further adjustments aim at transferring some vertices from the clusters with sizes larger than $S_{\text{max}}$ to the clusters with sizes smaller than $S_{\text{max}}$. Then the aforementioned issue is addressed. To begin with, the mean values of QoS requirements and path losses in cluster $C_i$ are respectively defined as

$$\bar{R}_{C_i} = \frac{\sum_{k \in C_i} R_k}{|C_i|} \quad (15)$$

and

$$\bar{PL}_{C_i} = \frac{\sum_{k \in C_i, j \neq k} PL'_{kj}}{|C_i| (|C_i| - 1)}. \quad (16)$$

Then $\bar{R}_{C_i}$ and $\bar{PL}_{C_i}$ are normalized as $\tilde{R}_{C_i} = \frac{\bar{R}_{C_i}}{R_{\text{max}}}$ and $\tilde{PL}_{C_i} = \frac{\bar{PL}_{C_i}}{PL_{\text{max}}}$, respectively. Accordingly, if vertex $l$ leaves cluster $C_i$ and joins cluster $C_s$, we have

$$\begin{align*}
\left\{ \begin{array}{l}
\tilde{R}_{C_i \setminus \{l\}} = \frac{\sum_{k \in C_i \setminus \{l\}} R_k}{(|C_i| - 1) R_{\text{max}}} \\
\tilde{PL}_{C_i \setminus \{l\}} = \frac{\sum_{k \in C_i \setminus \{l\}, j \neq k} PL'_{kj}}{(|C_i| - 1) (|C_i| - 2) PL_{\text{max}}} \\
\tilde{R}_{C_s \cup \{l\}} = \frac{\sum_{k \in C_s, j \neq l} R_k}{(|C_s| + 1) R_{\text{max}}} \\
\tilde{PL}_{C_s \cup \{l\}} = \frac{\sum_{k \in C_s, j \neq l, k \neq l} PL'_{kj}}{|C_s| (|C_s| + 1) PL_{\text{max}}}.
\end{array} \right. \quad (17)
\end{align*}$$

The adjustments will have effects on the similarities in both QoS requirements and path losses within the formed clusters, which have already been maximized after Algorithm 1. Thus, to have the slightest effects on the similarities within the formed clusters, the vertex which will leave cluster $C_i$ ($|C_i| > S_{\text{max}}$) is determined by

$$l^* = \arg\min_{l \in C_i, C_s \subseteq K_{\text{max}}} \left| \tilde{R}_{C_i \setminus \{l\}} \cdot \tilde{PL}_{C_i \setminus \{l\}} - \tilde{R}_{C_i} \cdot \tilde{PL}_{C_i} \right|, \quad (18)$$

where $K_{\text{max}}$ is the set of clusters with sizes larger than $S_{\text{max}}$. Then the cluster which vertex $l^*$ will join $C_s$ if its leaving and joining have the slightest effects on $\tilde{R}_{C_i} \cdot \tilde{PL}_{C_i}$ and $\tilde{R}_{C_s} \cdot \tilde{PL}_{C_s}$, respectively.

Moreover, the weight of cluster $C_i$ is defined as

$$W_{C_i} = \tilde{R}_{C_i} \cdot \tilde{PL}_{C_i}. \quad (20)$$

There is no systematic or theoretic research on how to choose the form of similarity function [30]. Due to its relative simplicity and neutrality, we use the exponential function as the similarity function, which was also used in [29].
Algorithm 2 Further Adjustments of Cluster Sizes

1. Initialize $K_{IA} = \emptyset$, $K_{more} = \emptyset$, and $K_{less} = \emptyset$.
2. if $|C_i| > S_{max}, \forall C_i \subseteq K_{IA}$
3. Let $K_{more} = K_{more} \cup C_i$.
4. if $K_{more} = \emptyset$
5. Turn to Algorithm 3.
6. end if
7. else if $|C_i| \leq S_{max} - 1$
8. Let $K_{less} = K_{less} \cup C_i$.
9. else Let $K_{IA} = K_{IA} \cup C_i$.
10. end
11. end if
12. while $|K_{more}| > 0$
13. for each $K_{leave} \subseteq K_{more}$
14. while $|C_{leave}| > S_{max}$ do
15. Determine SUE $t^*$ in $C_{leave}$ and $C_{join}$ which SUE $t^*$ will join according to (18) and (19), respectively.
16. Let $C_{leave} = C_{leave} \setminus \{t^*\}$ and $C_{join} = C_{join} \cup \{t^*\}$.
17. if $|C_{join}| = S_{max}$
18. Let $K_{less} = K_{less} \cup C_{join}$ and $K_{IA} = K_{IA} \cup C_{leave}$.
19. end if
20. end while
21. Let $K_{more} = K_{more} \setminus C_{leave}$ and $K_{IA} = K_{IA} \cup C_{leave}$.
22. end for
23. end while
24. Let $K_{IA} = K_{IA} \cup K_{less}$.
25. Sort all the $\Gamma$ clusters by the ascending order of their weights defined in (20).

Since both QoS requirements and path losses are similar in each cluster, a cluster with smaller weight indicates that all the SUEs have lower QoS requirements in this cluster, so less subchannels will be needed to guarantee the QoS requirements for most of SUEs in the cluster; moreover, the path losses are much similar in this cluster, so it is of great necessity to eliminate the strong intra-cluster interference by IA. As a result, to accord with the objective function in (9), the clusters with smaller weights should be given priority to subchannel allocation. This explains why the clusters are sorted by the ascending order of their weights in step 25 of Algorithm 2. Finally, we obtain the set of $\Gamma$ disjoint clusters with unequal sizes sorted by the ascending order of their weights, which is denoted as $K_{IA} = \{C_1, \ldots, C_{\Gamma}\}$.

**Lemma 1:** Not all the disjoint $\Gamma$ clusters formed after Algorithm 2 have equal sizes.

**Proof:** We assume that the sizes of all the disjoint $\Gamma$ clusters formed after Algorithm 2 are equal, i.e., $|C_1| = |C_2| = \ldots = |C_{\Gamma}| = S_0$, where $\Gamma > 1$, $S_0 \leq S_{max}$, and $\Gamma$, $S_0 \in \mathbb{N}^+$. Following this assumption, we have $\Gamma \cdot S_0 = K$, which can always be met only when $\Gamma = 1$ and $S_0 = K$, or $\Gamma = K$ and $S_0 = 1$, since the number of SUEs $K$ is an arbitrary positive integer. However, when $\Gamma = 1$ and $S_0 = K$, all the $K$ SUEs form only one cluster where the feasibility condition of IA cannot be guaranteed; when $\Gamma = K$ and $S_0 = 1$, each cluster contains only one SUE, which makes it impossible to perform IA in each cluster. So the initial assumption is contradicted, which implies that there exists at least one cluster whose size is not equal to those of other $\Gamma - 1$ clusters. This completes the proof of Lemma 1.

C. Phase 3: Subchannel Allocation for Clusters

In this phase, the subproblem of subchannel allocation for clusters becomes

$$\max_{\rho_{C_i}^m} \|q\|_1 \quad (21)$$

s.t. $C_4 - C_8$,

which is also NP-hard due to the combinatorial nature of subchannel allocation. Obtaining its optimal solution by exhaustive method will incur unaffordable computational burden on HeNB GW. Besides, due to constraints C5 and C6, the exhaustive method necessitates the estimation of global CSI, which will result in huge signaling overhead in dense SCNs. Therefore, greedy-style efficient algorithms\(^6\) with low-complexities are proposed to solve this subproblem, which can also greatly reduce the signaling overhead of CSI estimation.

To begin with, we define the path loss between arbitrary two clusters $C_i$ and $C_s$ as the minimum value between path losses from all SBSs in $C_i$ to all SUEs in $C_s$ and those from all SBSs in $C_s$ to all SUEs in $C_i$, i.e., $PL_{C_i,C_s} = \min \left\{ \{PL_k|k \in C_s, l \in C_s\} , \{PL_{kl}|l \in C_s, k \in C_i\} \right\}$. Then, if $PL_{C_i,C_s} > PL_0$\(^7\), the inter-cluster interference caused by SBSs in $C_i (C_s)$ at each SUE in $C_s (C_i)$ over each subchannel will not exceed the given threshold, so $C_i$ can share the same subchannels with $C_s$; otherwise, orthogonal subchannels will be allocated to them to mitigate the inter-cluster interference. Then $C_6$ is relaxed to the following constraint:

$$C_6': \rho_{C_i}^m \cdot \rho_{C_s}^m = \begin{cases} 1, & \text{if } PL_{C_i,C_s} > PL_0, \forall C_i, \forall C_s, \forall n. \\ 0, & \text{otherwise} \end{cases} \quad (22)$$

Similarly, we define the path loss from an arbitrary cluster $C_i$ to an arbitrary MUE $m$ as the minimum value of path losses from all SBSs in $C_i$ to MUE $m$, i.e., $PL_{m,C_i} = \min \{|PL_{mk}|k \in C_i\}$. Then, if $PL_{m,C_i} > PL_0$, the cross-tier interference caused by SBSs in $C_i$ at MUE $m$ over each of its allocated subchannels will not exceed the given threshold, either, so $C_i$ can reuse the subchannels allocated to MUE $m$; otherwise, it is not allowed to do so. Then $C_5$ is relaxed to the following constraint:

$$C_5': \rho_{C_i}^m = \begin{cases} 1, & \text{if } PL_{m,C_i} > PL_0, \forall m, \forall C_i, \forall n_m. \\ 0, & \text{otherwise} \end{cases} \quad (23)$$

Through the relaxations above, the estimation of accurate CSI is only confined within each cluster, which greatly reduces the signaling overhead. In a practical system, the accurate CSI estimation in cluster $C_i$ over subchannel $n$ can be implemented by allowing different SBSs in $C_i$ to transmit pilot symbols over

\(^6\)This phase requires to perform IA in each cluster over each subchannel. Each SBS plays a role of transmitter, and its served SUE plays a role of receiver. So each SBS-SUE pair cannot be treated as a vertex in the proposed algorithms any more.

\(^7\)In dense SCNs, the path loss from SBS $k$ (j) to SUE $j$ (k) is larger than that to its served SUE $k$ (j), so the path loss between vertices $k$ and $j$ is larger than $PL_0$, i.e., $PL_{kj} > PL_0$ we think that $PL_{jk} > PL_0$ and $PL_{kj} > PL_0$. 


different subcarriers\(^8\) of subchannel \(n\) and different antennas at each SBS to transmit pilot symbols in different time slots, respectively. After receiving the pilot symbols from different transmit antennas at different SBSs over each subcarrier of subchannel \(n\), each receive antenna at each SUE estimates the corresponding channel impulse responses (CIRs) through least square (LS) algorithm. Then the CIRs over remaining subcarriers of subchannel \(n\) can be obtained by linear interpolation. After finishing the CSI estimation in each cluster, each SUE reports the estimated accurate CSI to its serving SBS. Then the cluster head\(^9\) of each cluster gathers the estimated CSI from the other SBSs in the same cluster through X2 interface (wired backhaul) [14], and then forwards the gathered CSI to HeNB GW through S1 interface.

Solving subproblem (21) also requires to estimate the path loss from each SBS to each MUE. The process is similar to that in the first phase. The only difference is that when it is finished, each MUE reports the estimated path losses to MBS, then MBS forwards them to HeNB GW.

The proposed algorithms consist of two procedures. The first procedure allocates one subchannel to each cluster, which is for constraint C4. The second procedure allocates the remaining subchannels to enhance SUEs’ sum rate if all the SUEs have satisfactory QoS requirements after the first phase; otherwise it allocates the remaining subchannels to the clusters containing SUEs with unsatisfactory QoS requirements. Details of the first and second procedures are shown in Algorithm 4 and Algorithm 5, respectively. Furthermore, constraint C5’ is guaranteed in steps 4-8 of Algorithm 3, and constraint C6’ is guaranteed in steps 2-8 of Algorithm 4, steps 5-7 of Algorithm 5 and steps 3-9 of Algorithm 6.

Next, we explain other new notations used in the proposed algorithms. \(\mathcal{C}_{IA}^*\) denotes the set of clusters, in each of which all SUEs have satisfactory QoS requirements, where \(\mathcal{C}_{IA}^* \subseteq \mathcal{C}_{IA}\). \(\mathcal{N}_{\mathcal{C}_i}^{avail}\) is the set of subchannels available for \(\mathcal{C}_i\) only under constraint C5’, i.e., it consists of subchannels that have already been allocated to the MUEs to whom the path losses from \(\mathcal{C}_i\) are larger than \(PL_0\). \(\mathcal{C}_{IA}^*\) denotes the set of clusters, to each of which the path loss from cluster \(\mathcal{C}_i\) is smaller than \(PL_0\). \(\mathcal{Q}_{\mathcal{C}_i}^{C_{i,1}}\) is the set of QoS guaranteed SUEs in cluster \(\mathcal{C}_i\) over subchannel \(n_{i,1}\) in the first procedure, and \(\mathcal{N}_{\mathcal{C}_i}\) denotes the set of subchannels finally allocated to \(\mathcal{C}_i\). \(\mathcal{Q}_{\mathcal{C}_i}^{N_{\mathcal{C}_i}}\), is the set of QoS guaranteed SUEs in \(\mathcal{C}_i\) over its allocated subchannels in \(\mathcal{N}_{\mathcal{C}_i}\). It is worth mentioning that under constraints C5’ and C6’, cluster \(\mathcal{C}_i\) may still achieve the maximum number of QoS guaranteed SUEs over multiple subchannels during the first procedure and each iteration of the second procedure. So we denote the set of these multiple subchannels by \(\mathcal{N}_{\mathcal{C}_i,u}\), where \(|\mathcal{N}_{\mathcal{C}_i,u}| \geq 1\) and \(u = 1,2\). All the possible cases that may happen during the whole process of subchannel allocation are summarized and the corresponding solution to each case is given as follows.

**Case 1.** In the first procedure, when we allocate one subchannel to the first cluster \(\mathcal{C}_1\), there will be no inter-cluster interference between \(\mathcal{C}_1\) and the other \(\Gamma - 1\) clusters, since the other \(\Gamma - 1\) clusters have not been allocated with any subchannels yet. In this case, all subchannels in \(\mathcal{N}_{\mathcal{C}_1}^{avail}\) are available for cluster \(\mathcal{C}_1\).

**Case 2.** In the first procedure, cluster \(\mathcal{C}_i\) achieves the maximum number of QoS guaranteed SUEs over subchannels in \(\mathcal{N}_{\mathcal{C}_i,1}\). So we select one subchannel from \(\mathcal{N}_{\mathcal{C}_i,1}\), according to the following criterion:

\[
n_{i,1}^* = \underset{n_{i,1} \in \mathcal{N}_{\mathcal{C}_i,1}}{\arg\max} \sum_{k \in \mathcal{C}_i} R_{k}^{n_{i,1}} = \arg\max \sum_{k \in \mathcal{C}_i} R_{k}^{n_{i,1}}.
\]

Note that this criterion selects a subchannel over which the achievable rates of SUEs in \(\mathcal{C}_i\) are closest to their QoS requirements to allocate to \(\mathcal{C}_i\), which efficiently utilizes the subchannel resources.

**Case 3.** In the first procedure, none of the SUEs in \(\mathcal{C}_i\) have satisfactory QoS requirements. In this case, one subchannel will be selected to be allocated to \(\mathcal{C}_i\) according to

\[
n_{i,1}^* = \underset{n_{i,1} \in \mathcal{N}_{\mathcal{C}_i}^{avail}}{\arg\max} \sum_{k \in \mathcal{C}_i} R_{k}^{n_{i,1}},
\]

which makes SUEs in \(\mathcal{C}_i\) try their best to achieve the data rates closest to their QoS requirements.

**Case 4.** After the first procedure, all the SUEs’ QoS requirements are guaranteed. So the remaining subchannels are allocated to enhance the sum rate of all SUEs.

**Case 5.** After the first procedure, some clusters still contain SUEs with unsatisfactory QoS requirements. In this case, Algorithm 6 is used to allocate the remaining subchannels to these clusters. Also, the following two sub-cases may happen during each iteration of Algorithm 6.

**Case 5a.** After one more subchannel is allocated to cluster \(\mathcal{C}_i\), the maximum number of QoS guaranteed SUEs in \(\mathcal{C}_i\) is the same as that before this one subchannel is allocated to \(\mathcal{C}_i\). So we select one subchannel from \(\mathcal{N}_{\mathcal{C}_i}^{avail}\) to allocate to \(\mathcal{C}_i\) according to

\[
n_{i,2}^* = \underset{n_{i,2} \in \mathcal{N}_{\mathcal{C}_i}^{N_{\mathcal{C}_i}}}{{\arg\max}} \sum_{k \in \mathcal{C}_i} \left( \sum_{n_{i} \in \mathcal{N}_{\mathcal{C}_i}} R_{k}^{n_{i,1}} \right) + R_{k}^{n_{i,2}},
\]

**Case 5b.** After one more subchannel is allocated to \(\mathcal{C}_i\), the maximum number of QoS guaranteed SUEs in \(\mathcal{C}_i\) is larger than that before this subchannel is allocated to \(\mathcal{C}_i\). In this case, we select one subchannel from \(\mathcal{N}_{\mathcal{C}_i,2}\) to allocate to \(\mathcal{C}_i\) according to

\[
n_{i,2}^* = \underset{n_{i,2} \in \mathcal{N}_{\mathcal{C}_i,2}}{\arg\max} \sum_{k \in \mathcal{C}_i} \left( \sum_{n_{i} \in \mathcal{N}_{\mathcal{C}_i}} R_{k}^{n_{i,1}} \right) + R_{k}^{n_{i,2}}.
\]

V. COMPUTATIONAL COMPLEXITY ANALYSIS

In this section, we analyze the computational complexities of the optimal solution and proposed solution. Finding the optimal solution to (9) requires to respectively exhaust all the possible cases of clustering and subchannel allocation. Firstly,
Algorithm 3 Efficient Subchannel Allocation for Clusters

1: Initialize $q = 0_{1 \times K}$ and $K^{IA}_1 = \emptyset$.
2: for $i = 1 : \Gamma$ do
3:   Set $N_c^{\text{avail}} = 0$, $I_c = \emptyset$, $N_c^{\text{avail}, 1} = 0$, $N_c = \emptyset$, $Q_{C_i} = \emptyset$ and $N_c^{\text{avail}, 2} = 0$, where $u = 1, 2$.
4:   for each MUE $m$ do
5:     if $PL_{c, m} > PL_0$ then
6:       $N_c^{\text{avail}} = N_c^{\text{avail}} \cup N_{m}^c$.
7:     end if
8:   end for
9:   if $\sum_{n_c^{\text{avail}}} R_k^{n_c^{\text{avail}}} \geq R_k, \forall k \in C_i$, $\forall C_i \subseteq K^{IA}_i \backslash \hat{K}^{IA}_i$ then
10:     Set $q_k = 1$.
11:   end if
12: end for
13: end for
14: for $i = 1 : \Gamma$ do
15:   Execute Algorithm 4 to allocate one subchannel to each cluster.
16: end for
17: Execute Algorithm 5 to allocate the remaining subchannels.
18: if $\sum_{n_c^{\text{avail}}} R_k^{n_c^{\text{avail}}} \geq R_k, \forall k \in C_i$, $\forall C_i \subseteq K^{IA}_i \backslash \hat{K}^{IA}_i$ then
19:     Set $q_k = 1$.
20: end if

Algorithm 4 First Procedure: Allocating One Subchannel to Each Cluster

1: for $i = 1 : \Gamma$ do
2:   Case 1: if $i = 1$, turn to step 5.
3:     else for $s = 1 : i - 1$
4:       if $PL_{c, m} = 0$ and $n_{s}^{\text{avail}, 1} \in N_{c, s}^{\text{avail}}$
5:         $N_c^{\text{avail}, 1} = N_c^{\text{avail}, 1} \cup \{n_{s}^{\text{avail}, 1}\}$.
6:     end if
7:   end for
8: Case 2: if $\max_{n_{s}^{\text{avail}, 1}} |Q_{C_i}^{n_{s}^{\text{avail}, 1}}| \geq 1$
9:     Let $N_c^{\text{avail}, 1} = \arg\max_{n_{s}^{\text{avail}, 1}} |Q_{C_i}^{n_{s}^{\text{avail}, 1}}|$ and $n_{s}^{\text{avail}, 1} = \arg\min_{n_{s}^{\text{avail}, 1} \in N_{c, s}^{\text{avail}, 1}} |\sum_{k \in C_i} R_k^{n_{s}^{\text{avail}, 1}}|$
10:     if $|n_{s}^{\text{avail}, 1}| = |C_i|$ then
11:       Set $K^{IA}_i = K^{IA}_i \cup C_i$ and $\Gamma = \Gamma - 1$.
12:     end if
13:     else Let $n_{s}^{\text{avail}, 1} = \arg\max_{n_{s}^{\text{avail}, 1} \in N_{c, s}^{\text{avail}, 1}} \sum_{k \in C_i} R_k^{n_{s}^{\text{avail}, 1}}$.
14: end if
15: Set $N_c^{\text{avail}} = N_c^{\text{avail}} \cup \{n_{s}^{\text{avail}, 1}\}$ and $N_c^{\text{avail}, 2} = N_c^{\text{avail}, 2} \backslash \{n_{s}^{\text{avail}, 1}\}$.
16: end for

the number of all clustering results, each of which consists of $\Gamma$ clusters with sizes not exceeding $S_{\max}$, is

$$\Phi_1 = \sum_{\sum_{i=1}^{\Gamma} |C_i| = K \atop \sum_{|C_i| \leq S_{\max}}} \left( \frac{K}{|C_i|} \right) \prod_{i=1}^{\Gamma - 1} \left( \frac{K - \sum_{i=2}^{\Gamma} |C_i|}{|C_i|} \right)^{\Gamma !}.$$  \hspace{1cm} (28)

And each cluster will be allocated with at least one and at most $N$ subchannels, so the number of all possible results of allocating $N$ subchannels to $\Gamma$ clusters is given by

$$\Phi_2 = \sum_{1 \leq |C_i| \leq N} \left\lfloor \prod_{i=1}^{\Gamma} \left( \frac{N}{|C_i|} \right)^{|C_i|} \Gamma ! \right\rfloor.$$  \hspace{1cm} (29)

where the complexity of designing the precoding and interference suppression matrices for IA in each cluster over each subchannel is simply considered as $\Psi$ [21]. So the computational complexity of the optimal solution is $O(\Phi_1 \Phi_2)$, which is unaffordable on HeNB GW.

In Algorithm 1, step 3 needs to compute the $\Gamma$ smallest eigenvalues of the sparse, large and symmetric matrix $L$ norm by Lanczos algorithm, which requires a complexity of $O(KT_1)$ [29], where $T_1$ is the number of iterations taken by Lanczos algorithm to converge. Moreover, step 5 requires a complexity of $O(KT_2)$ [32], where $T_2$ is the number of iterations taken by k-means algorithm to converge. Since $O(KT_1)$ and $O(KT_2)$ are smaller than $O(KTT_2)$, Algorithm 1
has a computational complexity lower than $O(K^2 T_1 T_2)$. In Algorithm 2, the worst case of steps 12-23 is that one cluster contains $K - (\Gamma - 1)$ SUEs, and each of the other $\Gamma - 1$ clusters contains only 1 SUE; in this case, steps 12-23 have a complexity lower than $O((\Gamma - K)^2)$. So the computational complexity of Algorithm 2 is lower than $O((\Gamma K)^2)$.

In Algorithm 3, steps 2-14 have a complexity of $O(K^2)$. Then, in Algorithm 4, steps 3-8 have a complexity lower than $K^2$. Furthermore, we assume that both case 2 and case 3 happen with probability $\frac{1}{2}$, and each has a complexity lower than $O(N^2)$. Therefore, the overall complexity of Algorithm 4 is lower than $O((\Gamma N)^2)$. In Algorithm 5, steps 3-7 require a complexity lower than $O((\Gamma N)^2)$, so steps 2-8 have a complexity lower than $O(\Gamma^2 N^2)$. In Algorithm 6, steps 4-8 have a complexity lower than $O(K^2)$. Thus, the overall complexity of Algorithm 5 is $O(\Gamma^2 N^2)$, and the overall complexity of Algorithm 3 is considered as $O(\Gamma^2 N^2)$. So the overall complexity of the proposed solution is lower than $O(KTT_1 T_2 + \Gamma^2 N^2)$, which is also much lower than that of optimal solution.

VI. NUMERICAL RESULTS

In the simulation, all SBSs overlaying a macrocell are deployed in a single-floor and square indoor area, and the locations of all SBSs are drawn from an independent homogeneous Poisson point process (PPP) with density $\lambda = \frac{K}{\pi R^2}$, where $EL$ is the edge length of square indoor area. All the SUEs’ QoS requirements are generated from a Gaussian distribution $\mathcal{N}\left(\mu_0, \sigma_0^2\right)$, where $\mu_0$ and $\sigma_0$ are the mean value and the standard deviation of all the SUEs’ QoS requirements, respectively. Other system parameters in simulation are given in Table 1. Note that the maximum transmit power of each SBS is set to 30 mW in [1] and [26], here we use half of this value as $p$. All of the SBSs and SUEs are equipped with 3 antennas, and each SBS sends 1 data stream to its served SUE. According to (3), we have $S_{\max} = 5$. However, there will be no closed forms of precoding and interference suppression matrices for IA if the number of users performing IA is more than 3 [33]. So we use the algorithm proposed in [33] and the formula in [15] to obtain the precoding and interference suppression matrices, respectively.

The indoor channel models (dual stripe model) in [34] are used to model the propagation environment, and the path loss from SBS $k$ to its served SUE $j$ is $PL_{jk}(dB) = 38.46 + 20\log_{10}d_{jk}$, where $d_{jk}$ is the distance from SBS $k$ to its served SUE $j$. In a given time slot of resource allocation, each SBS serves 1 SUE randomly located in a circular disc around this SBS with an inner radius and an outer radius of 3 m and 10 m [34], respectively. Therefore, $PL_0 = 38.46 + 20\log_{10}10 = 58.5$ dB. Besides, the path loss from SBS $k$ to SUE $j (k \neq j)$ does not serve is $PL_{jk}(dB) = 38.46 + 20\log_{10}d_{jk} + \delta_1 L_{iw}$, where $\delta_{jk}$ denotes the distances from SBS $k$ to SUE $j$ it does not serve, and $\delta_1$ is the number of inner walls between SBS $k$ and SUE $j$, and $L_{iw}$ is the penetration losses due to an inner wall, which equals 5 dB. And for the path loss from SBS $k$ to the outdoor MUE $m$, we have $PL_{mk}(dB) = \max(15.3 + 37.6\log_{10}d_{mk}, 38.46 + 20\log_{10}d_{mk}) + \delta_2 L_{ow}$, where $d_{mk}$ represents the distances from SBS $k$ to the outdoor MUE $m$, and $\delta_2$ is the number of inner walls between SBS $k$ and MUE $m$, and $L_{ow}$ denotes the penetration losses due to an outdoor wall whose value is 20 dB. Each MUE has been allocated with one subchannel before the subchannel allocation for SUEs. All the numerical results are obtained by averaging over 200 realizations. In each realization, the channel matrices, the positions of each SBS and SUE, and the QoS requirement for each SUE are varied. So the number of QoS guaranteed SUEs achieved by all the six schemes are not integers. For comparison, the performances of following six schemes are evaluated in simulation under various parameters. Note that obtaining the optimal solution to problem (9) by exhaustive search will incur prohibitive computational complexity in dense SCNs even when $N = 10$. Thus, as an efficient approximation of the optimal solution, Algorithm 1 and Algorithm 2 are used for clustering for SUEs, and exhaustive search is used for the optimal subchannel allocation for clusters (CSQPL+OSA).

1) CSQPL+OSA.
2) CSQPL. This is the proposed scheme in our paper.
3) Clustering Based on Similarity only in Path Losses (CSPL). This scheme requires $w_{kj} = \exp\left(-\frac{PL_{jk}}{PL_{mk}}\right)$, and Algorithm 1-Algorithm 6 are used to solve (9).
4) Clustering Based on Similarity only in SUEs’ QoS Requirements (CSQ). This scheme requires $u_{kj} = \exp\left(-\frac{R_{kj} - R_{\max}}{R_{\max} - R_{\min}}\right)$, and Algorithm 1-Algorithm 6 are used to solve (9).
5) Random Clustering. All the SUEs are randomly grouped into $\Gamma$ clusters, each of which contains at most $S_{\max}$ SUEs, and then Algorithm 3-Algorithm 6 are executed.
6) No Clustering. This scheme directly allocates the subchannels to the SUEs by the ascending order their QoS requirements, and each SBS sends 1 data stream to its served SUE along the largest singular value of the channel matrix between them.

Fig. 2 investigates the number of QoS guaranteed SUEs with respect to the number of SUEs. We have $\mu_0 = 1.5$ Mbps, $\sigma_0 = 0.5$ Mbps and $EL = 20$ m. When $K$ varies

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Carrier frequency</td>
<td>2 Ghz</td>
</tr>
<tr>
<td>Noise power density, $N_0$</td>
<td>-174 dBm/Hz</td>
</tr>
<tr>
<td>Subchannel bandwidth, $\Delta f$</td>
<td>0.18 MHz</td>
</tr>
<tr>
<td>Number of subchannels, $N$</td>
<td>10</td>
</tr>
<tr>
<td>Transmit power of each SBS, $p$</td>
<td>11.7 dBm (15 mW)</td>
</tr>
</tbody>
</table>
from 20 to 40 with a step length of 5, the corresponding values of λ are 0.05, 0.0625, 0.075, 0.0875 and 0.1. Firstly, CSQPL outperforms both CSPL and CSQ since it can utilize subchannels more efficiently. Moreover, the performances of CSQPL, CSPL and CSQ almost linearly increase with K, which are much better than that of no clustering scheme. The reason is that as K increases, the co-tier interference becomes stronger, and less subchannels are available to each cluster. However, at most 5 SUEs are contained in each cluster to perform IA, which leads to a much higher spectral efficiency compared with no clustering scheme. Besides, as K increases, the path losses from each SBS to the SUEs it does not serve become smaller, so the performance of CSQ approaches to that of CSPL and begins to outperform it when K > 35. We can also observe that the performance of CSQ gets close to that of CSQPL; however, CSQPL will always outperform CSQ because it considers similarities in both QoS requirements and path losses when grouping SUEs into disjoint clusters, while CSQ considers similarity only in QoS requirements. Actually, their performance curves become almost parallel when K ≥ 35. Finally, CSQPL has a smaller performance loss compared with CSQPL+OSA as K increases, but it has a notably reduced computational complexity.

Fig. 3 plots the number of QoS guaranteed SUEs with respect to the mean value of SUEs’ QoS requirements. We have K = 30, μ₀ = 1.5 Mbps, EL = 20 m and λ = 0.075. The increase in μ₀ implies that the SUEs’ QoS requirements become higher, so each cluster formed by the schemes of CSQPL, CSPL, CSQ and random clustering will need more subchannels to make SUEs in it have satisfactory QoS requirements. However, there are only 10 subchannels available, so the number of QoS guaranteed SUEs achieved by the aforementioned four schemes greatly decrease as μ₀ increases. Besides, each cluster formed by CSPL contains SUEs with random QoS requirements, which will lead to a lower spectral efficiency. This is because our goal is to maximize the number of QoS guaranteed SUEs, more subchannels are only needed by the SUEs with higher QoS requirements in a cluster, but are quite redundant for the SUEs with lower QoS requirements in this cluster as they have already had satisfactory QoS requirements after being allocated with less subchannels. So CSQPL outperforms CSPL, and CSQ also begins to outperform CSQ when μ₀ ≥ 1.9 Mbps. Moreover, the performance gap between CSQPL and CSQ becomes smaller when 1 Mbps ≤ μ₀ ≤ 2.5 Mbps, but CSQPL always exhibits a better performance than CSQ because it takes similarities in both QoS requirements and path losses into consideration in the phase of clustering. We can also observe that the performance of no clustering scheme degrades at a much slower pace, but it is still much worse than that of CSQPL.

Fig. 4 shows the number of QoS guaranteed SUEs with respect to the standard deviation of SUEs’ QoS requirements. We have K = 30, μ₀ = 1.5 Mbps, EL = 20 m and λ = 0.075. In CSQPL, a cluster with smaller weight is given priority to subchannel allocation, because less subchannels need to be allocated to this cluster to meet the QoS requirements for most of the SUEs in it. So more subchannels can be retained for the clusters with bigger weights, which implies that CSQPL can utilize the limited subchannel resources more efficiently. Therefore, CSQPL greatly outperforms other related schemes except the optimal solution as σ₀ increases. CSQPL is also able to adapt the increase in σ₀ (i.e., the fluctuation in SUEs’ QoS requirements) because its performance experiences mild degradation as σ₀ increases. Moreover, the performance of CSQ is getting close to that of CSQPL, but it can never be better than that of CSQPL since similarity only in SUEs’ QoS requirements is taken into account when clustering. Finally, the performance gap between CSQPL and the optimal solution becomes a little larger as σ₀ increases; nevertheless, CSQPL has an significantly reduced computational complexity.

Fig. 5 plots the number of QoS guaranteed SUEs with respect to the edge length of indoor area. We have K = 30, μ₀ = 1.5 Mbps and σ₀ = 0.5 Mbps. When EL varies from 15 m to 35 m, the corresponding values of λ are 0.1333,
0.075, 0.048, 0.0333 and 0.0245. We can observe that CSQPL outperforms other related schemes except CSQPL+OSA when $15 \leq EL < 35$. Besides, the performance gap between CSQPL and CSQ, and that between CSPL and CSQ become larger as $EL$ increases. This is because QoS requirements are similar but path losses are random in the clusters formed by CSQ, the inter-cluster interference still needs to be mitigated by orthogonal subchannel allocation. Nevertheless, for the clusters formed by CSQPL and CSPL, the inter-cluster interference becomes relatively weak as $EL$ increases, which can be mitigated by allowing disjoint clusters to reuse subchannels. No clustering scheme exhibits a performance with exponential increase when $15 \leq EL \leq 35$. Besides, without clustering for SUEs or performing IA, no clustering scheme has significantly reduced complexity; when $EL \geq 35$, it can achieve the same performance as that of CSQPL, so it is much more preferable to CSQPL.

VII. CONCLUSION

In this paper, we have proposed a centralized efficient subchannel allocation scheme, which exploits IA with similarity clustering to maximize the number of QoS guaranteed SUEs in dense SCNs underlaying a macrocell. Our formulated problem was a combinatorial optimization problem. Due to its NP-hardness, we have proposed a three-phases efficient solution with notably reduced computational complexity which consists of three phases: similarity clustering for SUEs through graph partitioning, further adjustments of cluster sizes to guarantee the feasibility condition of IA in each cluster and efficient subchannel allocation for the formed clusters. Also, the proposed solution greatly reduces the signaling overhead of CSI estimation. We also showed that the proposed solution has notably reduced computational complexity compared with that of the optimal solution. Simulation results demonstrated that the proposed solution not only outperforms other related schemes, but also achieves a close performance to the optimal solution.

REFERENCES


