Efficient mmWave Wireless Backhauling for Dense Small-Cell Deployments

Po-Han Huang and Konstantinos Psounis
Ming Hsieh Department of Electrical Engineering
University of Southern California, Los Angeles, CA, USA, 90089
Email: {pohanh, kpsounis}@usc.edu

***Abstract*—Dense small-cell deployments of 5G networks require a wireless backhaul to efficiently connect the small cells to the macro base station (BS). We envision a wireless backhaul architecture where cells are grouped into clusters. One small cell per cluster plays the role of a cluster head connecting the rest of the small cells to the macro cell via a mmWave MIMO link. We formulate the problem of jointly selecting the cluster heads and the number of BS antennas dedicated to each mmWave MIMO link between the BS and each cluster head as a mixed integer nonlinear program (MINLP) and prove its NP-hardness. We propose a Alternate Convex Search Heuristic (ACSH) to handle the tradeoff between having faster backhaul links versus having more cluster heads and show it is near-optimal via extensive simulations. Last, we show that our heuristic has a 20%-50% performance gain compared to prior work.**

***Keywords*—Wireless backhaul networks, mmWave communication, hybrid beamforming, dense small-cell deployments.**

I. Introduction

To meet the high demand in 5G cellular networks, several research directions are explored to increase the capacity of both the access networks and the backhaul networks [1]. For wireless access networks, one promising solution is to use a highly dense base station deployment, which could enhance the whole system throughput by frequency reuse. To support such a network deployment, the backhaul network should be re-designed because it would be too expensive to connect such a large number of access points via fibers [2].

MmWave communication has recently matured thanks to hardware design advancements, and has been proposed to support the high bandwidth demand in 5G cellular networks [3]. The rationale behind using mmWave communication is to take advantage of higher frequency bands, e.g. from 30 GHz to 300 GHz, which could provide higher capacity with larger bandwidth than todays microwave bands. For this reason, mmWave communication is considered suitable for high-bandwidth backhaul connection of ultra-dense small cells [4].

However, there are some fundamental challenges with mmWave communication, such as directivity challenges, high pathloss, low penetration and more [5]. These challenges make mmWave communication only useful for short-range transmissions. To make mmWave links handle long-range transmissions, it has been recently proposed to use MIMO beamforming [6]. This MIMO beamforming is not expected to incur significant inter-antenna interference thanks to the intrinsic directional nature of the transmissions which has led researchers to model mmWave backhaul links as pseudo-wires without interference [7].

Motivated by this, [8]-[19] discuss how to use mmWave communication in wireless backhaul networks. Some of this work focuses on a distributed architecture (e.g., [8]-[10]), where network operators deploy cluster heads connected with fibers to the core network and provide mmWave backhaul connection for the rest of the small cells. Another approach is to use a centralized architecture (e.g., [11]-[14]), where a central node like a macro cell controls every small cell via a mmWave backhaul. However, neither approach is particularly appealing for a dense network deployment. For example, the centralized approach has too high of a signaling complexity, and the distributed approach cannot handle well the fluctuation of traffic demand as the number of cluster heads is fixed.

To address those problems a hybrid architecture has been proposed, see, for example, [15]-[19], where a centralized node (e.g., macro cell) controls several cluster heads which are also small cells via mmWave communication, and these cluster heads provide wireless backhaul for the rest of the small cells. In this fully wireless backhaul network architecture, the macro cell only controls the cluster heads rather than all small cells, and there is great flexibility for changing cluster heads if the traffic demand fluctuates. Still, due to the sort range of mmWave communication, multi-hopping might be required, and, to avoid the performance issues of multi-hopping, MIMO beamforming has been suggested to support long-range transmissions between the macro cell and the cluster heads [6].

In the context of such a hybrid architecture with MIMO-enabled mmWave backhaul links, in this work we study how to optimally select cluster heads among the small cells and how to optimally select the link capacity of the backhaul links between the macro cell and the cluster cells, that is, how many antennas of the macro cell to dedicate at each backhaul link, to maximize the achieved system throughput. For this purpose, we formulate this problem into a mixed integer nonlinear program (MINLP) and prove its NP-hardness by reducing it to a k-set cover problem. To solve the problem in polynomial time, we first transform it into a simpler problem which ignores coverage constraints, and use an iterative algorithm to reach a near-optimal performance. Motivated by this approach we propose an Alternate Convex Search Heuristic (ACSH) to solve the original problem while satisfying both coverage and antennas constraints simultaneously and show its near-optimality by simulation.

The rest of this paper is organized as follows. Section II briefly summarizes the related work. Section III presents the system architecture that we consider. The objective of this work and a formal problem description is presented in Section IV. In Section V we propose the ACSH algorithm. In Section VI we evaluate the performance of the ACSH algorithm using simulations and show that it outperforms previous works under a variety of realistic scenarios. Last, Section VII concludes the paper.

II. Related Work

We summarize prior work on backhaul design focusing on distributed, centralized and hybrid architectures.

In [8]-[10] a distributed architecture with two types of nodes is presented: anchored nodes and demand nodes, where anchored nodes are used for backhaul relaying and demand nodes are used for serving users. The authors investigate how to select anchored nodes and where to place them while connecting them with wires for achieving high throughput. This prior work does not consider wireless backhauling.

A centralized architecture is investigated in [11]-[14]. The authors investigate how to efficiently allocate antennas of the macro cell to provide a wireless backhauling solution towards all small cells, and a number of works discuss the role of MIMO beamforming methods for providing better performance. This work does not consider the option to select a subset of small cells to act as relays (cluster heads) for the rest of the small cells.

[15] and [16] introduced and outlined the benefits and the drawbacks of a hybrid architecture where cluster heads are selected to serve small cells and are connected to a macro cell via mmWave links. Recent technical works on hybrid architectures, e.g. [17] and [18], have studied how to associate small cells with cluster heads subject to a predetermined number and location of cluster heads. This line of work does not consider the dynamic selection of cluster heads leading to dynamic network topologies which provide higher throughput under traffic fluctuations.

III. System Architecture

A. Hybrid Architecture for Wireless Backhaul

According to the hybrid architecture of wireless backhaul design for small cell networks (see Fig. 1), packets travel between the core network and a macro base station (BS) via fiber, then travel between the macro cell and a cluster head via mmWave communication, and last between a cluster head and the other small cell again via mmWave communication.

The main assumptions that we make are as follows. First, similar to prior work [7], we assume that the communication between cluster heads and the macro cell, as well as the communication between small cells and cluster heads are interference-free thanks to the use of beamforming. With this assumption there is no need to worry about interference between backhaul communication links. However, based on

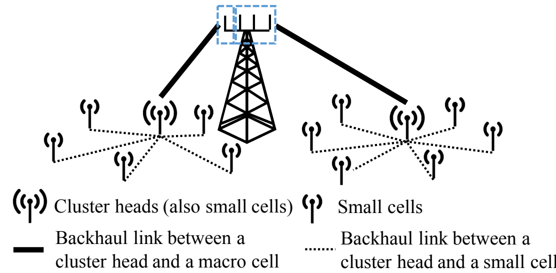


Fig. 1: Hybrid architecture.

[13], beamforming will suffer from alignment issues, especially in a mmWave environment. For this reason, rapidly updating cluster heads is impractical and, instead, we assume that cluster heads change at slow time scales, e.g. every hour or more, such that the alignment issues can be addressed by standard schemes like the one presented in [13].

Second, we assume that every small cell always has packets to send, i.e. we operate in the saturated throughput regime. With this assumption in mind, the amount of traffic between small cells and cluster heads is not the main focus because it is always confined by the link capacity between the cluster heads and the macro cell. The main goal is to select more cluster heads with higher backhaul capacity among the set of small cells under the constraints of coverage and maximum number of available antennas. Note that a small cell which becomes a cluster head should not only deal with the data from its users, but also with the traffic from the other small cells which use it as a relay.

Last, we select to allocate to each link between a cluster head and the macro cell the same capacity. This simplifies the analysis without masking the dynamics that we want to investigate, and, it is a reasonable assumption from a practical point of view considering that cellular providers attempt to load balance the traffic among their cells as much as possible.

B. Hybrid Beamforming via mmWave Communication

The beamforming model in this paper is based on the results in [19][20]. Note that this model works with different physical layer settings, and the main results of this paper won't be affected by the choice for such settings. The number of required antennas N_i to achieve backhaul capacity C if d_i is the distance between the macro cell and small cell i can been shown to equal [20]

$$N_i(C, d_i) = \left\lceil \frac{(2^{\frac{C}{W}} - 1) N_0 W d_i^{\alpha}}{P_t} \right\rceil, \tag{1}$$

where P_t is the transmission power, α is the path-loss coefficient, W is the bandwidth of a frequency slot, and N_0 is the noise power spectral density.

IV. Problem Formulation

In this section, we formulate the problem of **mm**Wave Wireless Backhauling for **H**ybrid **A**ccess **UL**tra-Dense Networks (mmHAUL) using mixed integer nonlinear program-

TABLE I: SYSTEM MODEL NOTATION

Description	Notation
Decision variable for determining which small cell i is cluster head	x_i
Decision variable for establishing connection between cluster head i and small cell j	$y_{i,j}$
The set of small cells	$\mathcal{S}$
Distance between macro cell and cluster head i	d_i
Transmission power of macro cell	P_t
Bandwidth of a frequency slot	W
Noise power spectral density	N_0
Path loss exponent	α
Backhaul capacity for every cluster head	C
Number of antennas for backhauling of cluster head i	$N_i(C, d_i)$
Maximum number of available antennas in macro cell	N_{MAX}
Adjacency matrix and its elements	$\mathcal{A} : a_{i,j}$

ming (MINLP). Specifically, we want to determine which small cells become cluster heads, which small cells connect to these cluster heads while maximizing the system throughput and ensuring connectivity for every small cell, and how many antennas of the macro cell are allocated to the connection between the macro cell and each cluster head. TABLE I lists useful notation.

A. Mixed Integer Nonlinear Programming (MINLP)

Given the set of small cells $\mathcal{S}$, the adjacency matrix

$$\mathcal{A} = \begin{bmatrix} a_{1,1} \cdots a_{1,|\mathcal{S}|} \\ \vdots \quad \ddots \quad \vdots \\ a_{|\mathcal{S}|,1} \cdots a_{|\mathcal{S}|,|\mathcal{S}|} \end{bmatrix},$$

which indicates whether small cell i and j are within range or not, and the number of available antennas at the macro cell N_{MAX}, we want to determine the values of the following decision variables: x_i which indicates whether small cell i is selected to be a cluster head or not, and $y_{i,j}$ which indicates whether small cell j connects to cluster head i, under the decision that the backhaul links between cluster heads and the macro cell will have the same capacity C, thus, cluster heads which are located further away from the macro cell would require more antennas N_i allocated to their link to support that capacity. We formulate this problem as follows:

$$Q1 : \max_{x_i, y_{i,j}, C} \sum\nolimits_{\forall i \in \mathcal{S}} C \cdot x_i \tag{2}$$

subject to

$$y_{i,j} \leq x_i \cdot a_{i,j},\ \forall i, j \in \mathcal{S} \tag{3}$$

$$\sum\nolimits_{\forall i \in \mathcal{S}} y_{i,j} = 1,\ \forall j \in \mathcal{S} \tag{4}$$

$$\sum\nolimits_{\forall i \in \mathcal{S}} N_i(C, d_i) \cdot x_i \leq N_{MAX}, \tag{5}$$

$$x_i = \{0, 1\},\ \forall i \in \mathcal{S}$$

$$y_{i,j} = \{0, 1\},\ \forall i, j \in \mathcal{S}$$

$$C \in \mathbb{R}+$$

The rationale behind this model is as follows. The objective (2) is to maximize the system throughput by either increasing the number of cluster heads $\sum_{\forall i \in \mathcal{S}} x_i$, which would result in a smaller link capacity C per cluster given the constraint on the total number of antennas N_{MAX}, or to increase C by selecting a smaller number of cluster heads which makes more antennas per cluster head available. (3) represents that the connection $y_{i,j}$ can be chosen if small cells i and j are adjacent to each other (i.e., $a_{i,j} = 1$) and small cell i has been chosen as a cluster head. (4) guarantees that every small cell connects to one cluster head. Last, (5) guarantees that the total number of used antennas for the links between cluster heads and the macro cell cannot exceed the number of available antennas at the macro cell. Optimizing the tradeoff between the total number of cluster heads and the capacity C of the backhaul link between the cluster heads and the macro cell (since, as already mentioned, more cluster heads implies less antennas per cluster and thus smaller C) is the main challenge in this framework.

B. Complexity Analysis

In this section, we show that mmHAUL is NP-hard by reducing it to a k-set cover problem. We start by defining the k-set cover problem and then we prove NP-hardness.

Definition 1. *(k-Set Cover Problem) Given a universe $\mathcal{U}$, an integer k, and a family $\mathcal{T}$ of subsets of $\mathcal{U}$, a cover is a subfamily $\mathcal{C} \subset \mathcal{T}$ of sets whose union is $\mathcal{U}$, and $|\mathcal{C}| \leq k$.*

Theorem 1. *mmHAUL is NP-hard*

Proof: Consider a fixed value of C. We can transform our problem to the standard form of the k-set cover problem by the following steps. (i) Because C is fixed, we can use $C' > C$ to render (2) into a minimization problem: $\min_{x_i, y_{i,j}} \sum_{\forall i \in \mathcal{S}} (C - C') \cdot x_i$. (ii) The constraints (3) and (4) are the typical coverage constraints in the k-set cover problem. (Note that (4) is usually expressed as a $\geq$ inequality rather than an equality but the solution is the same in our case.) (iii) Because C is fixed, the values of N_i are known. We replace in (5) the N_i's with the largest of them, say N_i' and obtain: $\sum_{\forall i \in \mathcal{S}} x_i \leq \frac{N_{MAX}}{N_i'}$, where $k = \frac{N_{MAX}}{N_i'}$. From this, it is evident that the k-set cover problem is a special case of our problem. Since the k-set cover problem is known to be NP-complete, our problem is NP-hard. ■

V. PROPOSED ALGORITHM

Since the above problem is too complicated to solve directly, in this section we propose an efficient heuristic to solve it. First, as a simplified example, we remove the coverage constraints (3) and (4) and the associated decision variable $y_{i,j}$ to obtain a simpler problem which we solve using the

Alternate Convex Search approach [21]. Then, following a similar procedure, we show how to solve the original problem.

A. mmHAUL without Coverage Constraints

In this subsection, we reformulate the original problem into a simpler problem by assuming that every small cell can reach every other small cell. Thus, (3), (4) and the associated decision variable $y_{i,j}$ can be removed. Therefore, we have the following simplified version of the original problem:

$$Q2 : \max_{x_i, C} \sum\nolimits_{\forall i \in \mathcal{S}} C \cdot x_i \tag{6}$$

subject to

$$\sum\nolimits_{\forall i \in \mathcal{S}} N_i(C, d_i) \cdot x_i \leq N_{MAX}, \tag{7}$$

$$x_i = \{0, 1\}, \ \forall i \in \mathcal{S}$$

$$C \in \mathbb{R}+$$

We call this problem $Q2$. For $Q2$, the main goal is to balance the backhaul link capacity and the number of cluster heads in order to maximize throughput under a constraint on the total number of available antennas. To solve $Q2$ we use a procedure similar to Alternate Convex Search [21]. Specifically, we first provide an initial solution $x_i, \forall i \in \mathcal{S}$, e.g. $x_i = 1 \ \forall i$, and solve the following subproblem to get the corresponding C:

$$Q3 : \sum\nolimits_{\forall i \in \mathcal{S} \& x_i = 1} \tilde{N}_i(C, d_i) = N_{MAX}, \tag{8}$$

where $\tilde{N}_i(C, d_i) = \frac{(2^{\frac{C}{W}} - 1) N_0 W d_i^{\alpha}}{P_t}$. In other words, $N_i(C, d_i) = \lceil \tilde{N}_i(C, d_i) \rceil$. We call this subproblem $Q3$.

Since $Q3$ is a rounding version of (7), if we use the C from $Q3$ into (7) the constraint may not be satisfied. To get a feasible solution based on this current C, we solve the following problem, called $Q4$, to get a feasible $x_i', \forall i \in \mathcal{S}$:

$$Q4 : \max_{x_i} \sum\nolimits_{\forall i \in \mathcal{S}} C \cdot x_i \tag{9}$$

subject to

$$\sum\nolimits_{\forall i \in \mathcal{S}} N_i(C, d_i) \cdot x_i \leq N_{MAX}, \tag{10}$$

$$x_i = \{0, 1\}, \ \forall i \in \mathcal{S}$$

$Q4$ is a standard Knapsack problem, and we can use a greedy algorithm [22] to get a near-optimal solution. Specifically, at each step we select the small cell i with the largest value of $R_i = \frac{C}{N_i}$ to become a cluster head, and subtract N_i from N_{MAX} until $N_{MAX} \leq 0$, see Algorithm 1 for more details.

With Algorithm 1 we obtain a new solution $x_i', \forall i \in \mathcal{S}$. We then put this solution into $Q3$ to obtain a new C', and use this new C' for $Q4$ to get a new solution with a new objective value. If the objective value is larger than or equal to the previous one (i.e., (6) with $x_i', \forall i \in \mathcal{S}$, and C), we keep solving $Q3$ and $Q4$ iteratively. When the performance cannot

Algorithm 1 Algorithm for $Q4$

Input: $N_i, \forall i \in \mathcal{S}$, C, and N_{MAX}.
Output: $x_i, \forall i \in \mathcal{S}$.

Initialize $x_i, \forall i \in \mathcal{S} = \{0, ..., 0\}$
Sort $\forall i \in \mathcal{S}$ by $R_i = \frac{C}{N_i}$ in descending order with new index $\tilde{i}$.
for $\forall \tilde{i} \in \mathcal{S}$ **do**
  if $N_{MAX} - N_{\tilde{i}} \geq 0$ **then**
    $x_{\tilde{i}} = 1$
    $N_{MAX} = N_{MAX} - N_{\tilde{i}}$
  end if
end for

Algorithm 2 Algorithm for $Q2$

Input: N_{MAX}.
Output: $x_i, \forall i \in \mathcal{S}$, and C.

Initialize a solution of $x_i', \forall i \in \mathcal{S}$, $O = 0$, and $O^* = 0$.
while $O \geq O^*$ **do**
  $O^* = O$
  $x_i = x_i', \forall i \in \mathcal{S}$
  Solve $Q3$ given $x_i, \forall i \in \mathcal{S}$ to obtain C.
  Solve $Q4$ given C to obtain $x_i', \forall i \in \mathcal{S}$.
  $O = \sum_{\forall i \in \mathcal{S}} C \cdot x_i'$
end while
$x_i = x_i', \forall i \in \mathcal{S}$

be further improved, the algorithm stops, see Algorithm 2 for more details.

It is worth mentioning that the complexity for solving $Q2$ is $O(c|\mathcal{S}| \log |\mathcal{S}|)$, where c represents the number of iterations until the algorithms stops, since the greedy algorithm for the Knapsack problem ($Q4$) requires $O(|\mathcal{S}| \log |\mathcal{S}|)$ operations. While there is no formal result bounding c, our simulation results indicate that in our problem setting it converges within a handful of iterations, see Figure 2a.

B. mmHAUL with Coverage Constraints

To solve mmHAUL, we extend the algorithm for $Q2$ by modifying some steps in Algorithm 2. Specifically, instead of solving $Q4$, we first make sure that the coverage constraint is satisfied by solving a new subproblem which we define below, and then we solve $Q4$ to maximize the objective function. Because this procedure resembles Alternate Convex Search (ACS) for biconvex optimization problems, we call our algorithm Alternate Convex Search Heuristic (ACSH).

The problem $Q5$ mentioned above is described as follows:

$$Q5 : \min_{x_i, y_{i,j}} \sum\nolimits_{\forall i \in \mathcal{S}} x_i \tag{11}$$

subject to

$$y_{i,j} \leq x_i \cdot a_{i,j}, \ \forall i, j \in \mathcal{S} \tag{12}$$

$$\sum\nolimits_{\forall i \in \mathcal{S}} y_{i,j} \geq 1, \ \forall j \in \mathcal{S} \tag{13}$$

$$\sum\nolimits_{\forall i \in \mathcal{S}} N_i(C, d_i) \cdot x_i \leq N_{MAX}, \tag{14}$$

$$x_i = \{0, 1\}, \ \forall i \in \mathcal{S}$$

$$y_{i,j} = \{0, 1\}, \ \forall i, j \in \mathcal{S}$$

The goal of $Q5$ is to use the minimum number of cluster heads/antennas to satisfy the coverage constraints (12) and (13). This is because we want to leave more available antennas after this step to have more options to achieve higher throughput by selecting more cluster heads when solving $Q4$. $Q5$ is a typical set-cover problem [22], and to solve it, at each step of a greedy procedure we select the small cell i with the smallest $K_i = \frac{N_i}{|\mathcal{S}_i|}$ to become the cluster head, where $\mathcal{S}_i$ is the set of small cells that can be covered by cluster head i, and $|\cdot|$ denotes the cardinality of that set. See Algorithm 3 for more details.

Algorithm 3 Algorithm for $Q5$

Input: $N_i, \forall i \in \mathcal{S}$.
Output: $x_i, \forall i \in \mathcal{S}$.

```
Initialize x_i, ∀i ∈ S = {0, ..., 0}.
Calculate |S_i|, ∀i ∈ S.
while |S|! = 0 do
    ĩ = arg min_{∀i∈S} K_i = N_i / |S_i|
    x_ĩ = 1
    S ← S − ĩ
    Update |S_i|, ∀i ∈ S.
end while
```

After solving $Q5$, we have $x'_i, \forall i \in \mathcal{S}$ that can cover all small cells. Then, we solve $Q4$ given C (obtained from solving $Q3$) and x'_i (obtained from solving $Q5$). Starting from the solution x'_i, the algorithm for $Q4$ will try to accommodate as many cluster heads as possible considering the available number of antennas. Like with the algorithm for $Q2$, ACSH will iteratively execute $Q3, Q5, Q4$ until the performance cannot be improved. Algorithm 4 shows a pseudo code for our main algorithm ACSH. The time complexity of ACSH is $O(c(|\mathcal{S}|^2 \log |\mathcal{S}| + |\mathcal{S}| \log |\mathcal{S}|))$, where c, like before, represents the number of iterations till the algorithm terminates. This is because the time complexity of the greedy algorithm of the set-cover problem requires $O(|\mathcal{S}|^2 \log |\mathcal{S}|)$ operations and of the Knapsack problem requires $O(|\mathcal{S}| \log |\mathcal{S}|)$ operations, and they will run c times each. Note that, like before, there is no formal bound for c, however, our simulations results show that a handful of iterations are enough for the algorithm to converge, see Figure 3a.

Algorithm 4 Alternate Convex Search Heuristic (ACSH) for $Q1$

Input: N_{MAX}.
Output: $x_i, \forall i \in \mathcal{S}$, and C.

```
Initialize a solution of x''_i, ∀i ∈ S, O = 0, and O* = 0.
while O ≥ O* do
    O* = O
    x_i = x''_i, ∀i ∈ S
    Solve Q3 given x_i, ∀i ∈ S to obtain C.
    Solve Q5 to obtain x'_i, ∀i ∈ S.
    Solve Q4 given C to obtain x''_i, ∀i ∈ S with the initial
    solution x'_i, ∀i ∈ S.
    O = Σ_{∀i∈S} C · x''_i
end while
x_i = x''_i, ∀i ∈ S
```

TABLE II: SIMULATION PARAMETERS

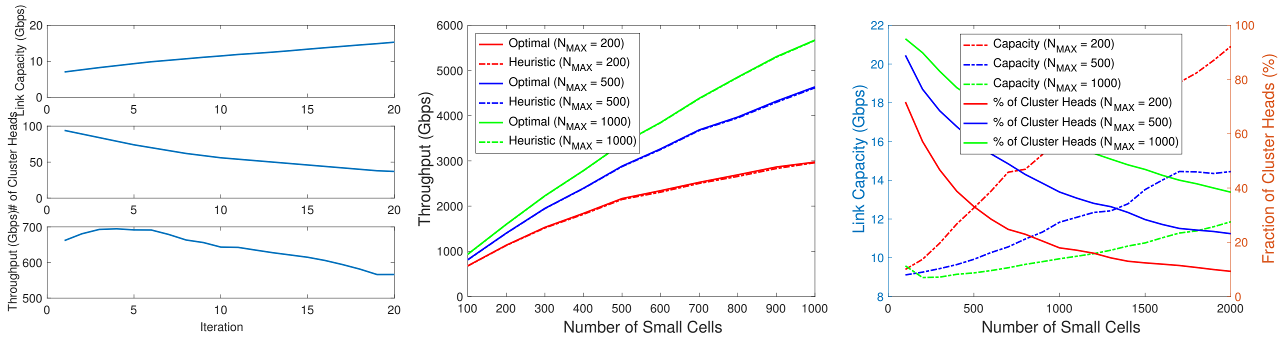
Parameters	Values
Radius of macro cell	500m
Transmission power of macro cell	30dBm
Bandwidth of a frequency slot	1GHz
Noise power spectrum density	3.2
Path loss exponent	5

VI. SIMULATION RESULTS

We implement our algorithms and the algorithms from prior work in MATLAB and CVX. We first show results related to $Q2$, including the performance of Algorithm 2 and the performance of the optimal solution for $Q2$, establishing by simulation that Algorithm 2 is near-optimality in a variety of network settings. Next, we present results related to the mmHAUL problem ($Q1$), comparing the performance of ACSH with that of the optimal solution for mmHAUL in small-scale networks, which shows that ACSH is near-optimal under the small scale scenarios we consider. Last, we study the performance of ACSH in large-scale networks and compare it with previous work in terms of system throughput, which shows that ACSH outperforms prior work.

A. Simulation Setting

The parameters used for the simulation are based on [9] and [19], and are listed in TABLE II. Small cells in the network are uniformly distributed in the range of the macro cell, which is 500m in large scale experiments and 200m in small scale experiments. The backhaul system is operated in 60GHz. The transmission power of the macro cell is 30dBm, and the bandwidth of a frequency slot is 1GHz. The path-loss coefficient is 5, and the noise figure equals 3.2. We study the system throughput under a varying number of small cells, a varying number of antennas on the macro cell, and varying small cell radius, to observe how these factors affect performance. Specifically, in large scale experiments we vary the number of small cells from 100 to 2000 and their radius from 100m to 500m (with 200m being the default value), and we vary the total number of antennas from 200 to 1000 to study systems with scarce, adequate, and abundant resources, respectively. And, in small scale experiments we vary the number of small cells from 10 to 50 having a default radius value of 100m, and we vary the total number of antennas from 20 to 100.



(a) System throughput v.s. number of iterations.

(b) System throughput v.s. number of small cells. (Comparison to optimal.)

(c) Link capacity and fraction of cluster heads v.s. number of small cells.

Fig. 2: Simulation results for mmHAUL without coverage constraints ($Q2$).

B. Results without coverage constraints

1) Number of Iterations: We set the number of small cells to 100 and the number of available antennas to 200 and study how many iterations are required for the algorithm to converge. We initialize all x_i to equal 1, making sure that the backhaul link capacity (C) starts from the lowest possible value, and increases at each step, which can be seen in the uppermost figure in Fig. 2a. With this initialization the number of cluster heads ($\sum_{\forall i \in \mathcal{S}} x_i$) starts from the highest value, and decreases at each step, which can be seen in the middle figure in Fig. 2a. The lowermost figure in Fig. 2a shows that the system throughput climbs up at the first few iterations, and is then drops down. Thus, Algorithm 2 terminates after a handful of iterations.

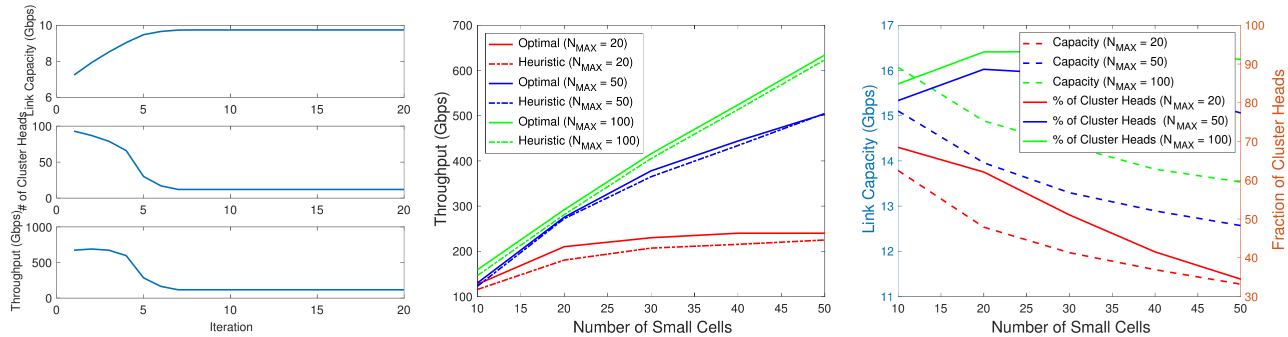
2) Effect of the Number of Small Cells: We examine the performance of Algorithm 2 under different network densities, i.e., different number of small cells in the network. As shown in Fig. 2b, Algorithm 2 performs almost as good as the optimal as the number of small cells increases and for all values of N_{MAX}. Also, as expected, the larger the number of available antennas the larger the achieved system throughput. Note that, while not visible with a bare eye, Algorithm 2 has larger discrepancy ($\sim$0.4%) with the optimal when N_{MAX} is 200 than when it assumes larger values. This is because the greedy algorithm for the Knapsack problem is farther from the optimal solution when the resources are more constrained, and, in our case, the resource is the number of available antennas. Last, as the number of small cells increases, more and more cells get within range of the macro cell leading to a larger load which increases the total throughput up to a point where the total number of antennas impose an upper bound, see, for example, the 200 antenna curve which begins to saturate.

Next, Fig. 2c plots the capacity of the backhaul links between cluster heads and the macro cell, as well as the proportion of small cells that are selected as cluster heads, as the number of small cells increase for a varying number of macro cell antennas. As the number of small cells increases the number of cluster heads increases as well, but, the *proportion* of small cells which are selected to be cluster heads decreases. Note that the total number of antennas imposes an upper bound on the number of cluster heads that we may select since each has to have at least one antenna dedicated to itself. Indeed, for say 200 antennas, the number of cluster heads grows up to 200 and then stays there as it cannot increase any further. At the same time, as the number of small cells increases the backhaul link capacity between cluster heads and the macro cell increases consistently with Fig. 2b.

C. Results with coverage constraints

1) Number of Iterations: We set the number of small cells to 100, the number of available antennas to 200 and study how many iterations are required for the ACSH algorithm to converge. We initialize all x_i to equal 1, thus, as before (Fig. 2a), the link capacity increases and the number of cluster heads decreases at every step (see the uppermost figure and the middle figure in Fig. 3a). The system throughput is ascending at the first iterations, and then is descending at the following iterations. This shows that ACSH will terminate after a few iterations. One difference between this figure and Fig. 2a is that these curves converge faster than those in Fig. 2a. This is because the solution space for $Q1$ is smaller than $Q2$ since $Q1$ has coverage constraints to be satisfied in addition to antenna constraints.

2) Effect of the Number of Small Cells in Small-scale Scenario: Because the complexity to solve the optimal solution of for $Q1$ increases exponentially with the number of small cells, we first use a small-scale network to compare ACSH versus the optimal. Specifically, as already mentioned in the simulation setting preamble, in this small scale scenario the radius of the macro cell is set to 200m, the radius of the small cells is 100m, we vary the number of small cells from 10 to 50, and we use 20, 50, and 100 antennas. As shown in Fig. 3b, ACSH is close to the optimal solution in all cases as the number of small cells increases. Moreover, as expected, ACSH achieves higher system throughput as the number of available antennas increases. Further note that the gap between our algorithm (ACSH) and the optimal is slightly larger now ($\sim$5%) than it was in Fig. 2b. We conjecture this occurs because the greedy set cover algorithm that we use to satisfy the coverage



(a) System throughput v.s. number of iterations.

(b) System throughput v.s. number of small cells. (Comparison to optimal.)

(c) Link capacity and fraction of cluster heads v.s. number of small cells.

Fig. 3: Small scale simulation results for mmHAUL with coverage constrains ($Q1$).

constraints may include as a cluster head a small cell that wouldn't be picked by the optimal algorithm. Note that with coverage constraints the system is forced to select some remote cells to be cluster heads to relay traffic from remote cells to the macro cell, leading to faster system throughput saturation.

Last, Fig. 3c plots the capacity of the backhaul links between cluster heads and the macro cell, as well as the proportion of small cells that are selected as cluster heads, as the number of small cells increase for a varying number of macro cell antennas. In contrast to Fig. 2c, coverage constraints cause the backhaul link capacities to decrease as the number of small cells increase, since a growing number of remote cluster heads are used to service remote small cells, and remote cluster heads require more antennas to achieve the same backhaul rates as cluster heads which are located closer to the macro cell. Also, like in Fig. 2c, the fraction of small cells which are cluster heads goes down since small cells increase but the number of cluster heads is bounded by the number of antennas.

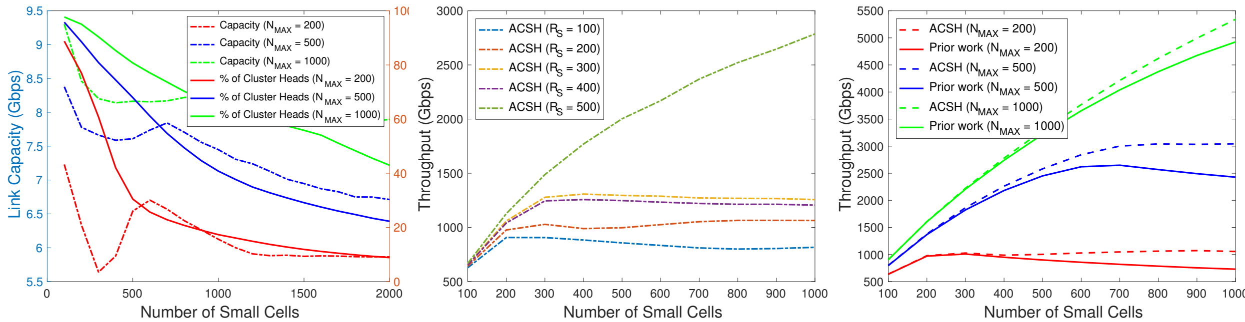
3) Effect of the Number of Small Cells in Large-scale Scenario: We conduct large-scale experiments with 100-2000 small cells, 200-1000 antennas at the macro cell, and 500m/200m radius for the macro/small cells respectively. Fig. 4a plots the capacity of the backhaul links between cluster heads and the macro cell, as well as the proportion of small cells that are selected as cluster heads, as the number of small cells increase for a varying number of macro cell antennas. The coverage constraints again force the system to select cluster heads which are relatively far from the macro cell such that traffic from remote small cells is relayed, resulting in a smaller backhaul link capacity as the number of small cells increases. Interestingly there is a local rebound on the link capacity before it goes down again. This happens when the number of cells are a bit larger than the number of antennas, because while the number of clusters heads can't be increased further, new cluster heads can be selected which are closer to the macro cell achieving higher link rates with the same number of antennas. Like before the number of cluster heads saturates due to antenna constraints and as the number of small cells keeps on increases the portion of small cells which are cluster heads goes down.

4) Effect of Small-Cell Radius: Fig. 4b plots the system throughput when 200 antennas are available, as a function of the number of small cells for a varying radius of small cells, R_S. As expected, the larger the radius of the small cells and thus of the cluster heads, the larger the system throughput, since the coverage constraint can be satisfied more easily allowing for more options to optimize the overall throughput. The $R_S = 500$ case corresponds to virtually no coverage constraints since all small cells can directly transmit to the macro cell, and the system throughput keeps on increasing like in Fig. 2b. In the rest of the cases coverage constraints force the system to select remote cluster heads to relay traffic from the increasing number of remote small cells, more antennas are required for those remote cluster heads, the system runs out of antennas and the throughput is saturated fast in contrast to Fig. 2b.

Last, by comparing the 200 antenna line in Fig. 2b with the $R_S = 500$ line in Fig. 4b (where there are virtually on coverage issues), we conclude that the ACSH algorithm, which solves subproblem $Q5$ before solving subproblem $Q4$, has a small system throughput penalty (around 5%) as compared to Algorithm 2 which is only concerned with solving subproblem $Q4$ (see the pseudo-codes in Section V).

5) Comparison with Prior Work: In this section, we compare ACSH with prior work. Specifically, we consider the state of the art hybrid approach from prior work presented in [17], where the authors solve a set coverage problem to guarantee coverage, but don't maximize the system throughput in a formal way. Instead, they preferentially select as cluster heads those small cells that cover/service as many small cells as possible.

Fig. 4c plots the performance of ACSH and prior work under a varying number of small cells, number of antennas, and a small cell radius of 200m. ACSH with $N_{MAX} = 200$ achieves a 0.2%-45.5% gain compared to prior work, and with $N_{MAX} = 500$ it achieves a 0.2%-24.8% gain. When the total number of antennas is equal to 1100, ACSH achieves a 0.1%-8.7% gain.



(a) Link capacity and fraction of cluster heads v.s. number of small cells.

(b) System throughput of ACSH with different small cell radius (R_S) v.s. number of small cells.

(c) System throughput of ACSH and related works v.s. number of small cells.

Fig. 4: Large scale simulation results for mmHAUL with coverage constrains ($Q1$).

VII. Conclusion

We propose to create a mmWave wireless backhauling network to connect small cells with a macro cell in the context of upcoming 5G networks. We formulate the problem of selecting some small cells to act as relays/cluster heads for other small cells, selecting the small cells to connect to each cluster heads, and selecting the number of macro cell antennas to be dedicated to the backhaul link between each cluster head and the macro cell as a mixed integer nonlinear program (MINLP) and prove it is NP-hard. We then proposal an algorithm called Alternate Convex Search Heuristic (ACSH) to efficiently solve it and study via simulation its performance.

References

[1] J. Rodriguez, Ed. *Fundamentals of 5G Mobile Networks*, 1st edition, John Wiley & Sons, 2015.

[2] Small Cell Forum, "Backhaul Technologies for Small Cells: Use Cases, Requirements, and Solutions," *Small Cell Forum Press Releases*, Feb. 2013.

[3] U. Siddique, H. Tabassum, E. Hossain, et al. "Wireless Backhauling of 5G Small Cells: Challenges and Solution Approaches," *IEEE Wireless Commun.*, vol. 22, no. 5, pp. 22-31, Oct. 2015.

[4] R. Baldemair, T. Irnich, K. Balachandran, et al. "Ultra-Dense Networks in Millimeter-Wave Frequencies," *IEEE Commun. Mag.*, vol. 53, no. 1, pp. 202-208, Jan. 2015.

[5] S. Sun, T. S. Rappaport, R. W. Heath Jr., et al. "MIMO for Millimeter-Wave Wireless Communications: Beamforming, Spatial Multiplexing, or Both?," *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 110-121, Dec. 2014.

[6] A. Adhikary, E. Al Safadi, M. K. Samimi, et al. "Joint Spatial Division and Multiplexing for mmWave Channels," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1239-1255, June 2014.

[7] S. Singh, R. Mudumdai, and U. Madhow, "Interference Analysis for Highly Directional 60-GHz Mesh Networks," *IEEE/ACM Trans. Netw.*, vol. 19, no. 5, pp. 1513-1527, Oct. 2011.

[8] E. Karamad, R. S. Adve, Y. Lostanlen, et al. "Optimizing Placements of Backhaul Hubs and Orientations of Antennas in Small Cell Networks," in *IEEE ICC'15 Workshop on BackNets*, London, UK, June 2015.

[9] S. Singh, M. N. Kulkarni, A. Ghosh, et al. "Tractable Model for Rate in Self-Backhauled Millimeter Wave Cellular Networks," *IEEE J. Sel. Area Commun.*, vol. 33. no. 10, pp. 2196-2211, Oct. 2015.

[10] X. Xu, W. Saad, X. Zhang, et al. "Joint Deployment of Small Cells and Wireless Backhaul Links in Next-Generation Networks," *IEEE Commun. Lett.*, vol. 19, no. 12, pp. 2250-2253, Dec. 2015.

[11] R. J. Weiler, M. Peter, W. Keusgen, et al. "Enabling 5G Backhaul and Access with Millimeter-Waves," in *IEEE EuCNC'14*, Bologna, Italy, June 2014.

[12] J. Zhao, T. Q. S. Quek, and Z. Lei, "Heterogeneous Cellular Networks Using Wireless Backhaul: Fast Admission Control and Large System Analysis," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 10, pp. 2128-2143, Oct. 2015.

[13] S. Hur, T. Kim, D. J. Love, et al. "Millimeter Wave Beamforming for Wireless Backhaul and Access in Small Cell Networks," *IEEE Trans. on Commun.*, vol. 61, no. 10, pp. 4391- 4403, Oct. 2013.

[14] H. Tabassum, A. Hamdi, and E. Hossain, "Analysis of Massive MIMO-Enabled Downlink Wireless Backhauling for Full-Duplex Small Cells," *IEEE Trans. on Commun.*, vol. 64, no. 6, pp. 2354-2369, June 2016.

[15] X. Ge, H. Cheng, M. Guizani, et al. "5G Wireless Backhaul Networks: Challenges and Research Advances," *IEEE Network*, vol. 28, no. 6, pp. 6-11, Nov/Dec 2014.

[16] K. Zheng, L. Zhao, J. Mei, et al. "10 Gb/s HetSNets with Millimeter-Wave Communications: Access and Networking - Challenges and Protocols," *IEEE Commun. Mag.*, vol. 53, no. 1, pp. 222-231, Jan. 2015.

[17] M. Z. Islam, A. Sampath, A. Maharshi, et al. "Wireless Backhaul Node Placement for Small Cell Networks," in *IEEE CISS'14*, Princeton, NJ, USA, Mar. 2014.

[18] Y. Zhu, Y. Niu, J. Li, et al. "QoS-aware Scheduling for Small Cell Millimeter Wave Mesh Backhaul," in *IEEE ICC'16*, Kuala Lumpar, Malaysia, May 2016.

[19] H. S. Dhillon and G. Caire, "Wireless Backhaul Networks: Capacity Bound, Scalability Analysis and Design Guidelines," *IEEE Trans. on Wireless Commun.*, vol. 14, no. 11, pp. 6043-6056, Nov. 2015.

[20] M. R. Akdeniz, Y. Liu, M. K. Samimi, et al. "Millimeter Wave Channel Modeling and Cellular Capacity Evaluation," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1164-1179, June 2014.

[21] J. Gorski, F. Pfeuffer, and K. Klamroth, "Biconvex Sets and Optimization with Biconvex Functions - A Survey and Extensions," *Mathematical Methods of Operations Research*, vol. 66, no. 3, pp. 373?407, Dec. 2007.

[22] D.-Z. Du, K.-I Ko, X. Hu, *Design and Analysis of Approximation Algorithms*, Springer, New York City, USA, 2012.