Coordination and Antenna Domain Formation in Cloud-RAN systems

Hadi Ghauch*, Muhammad Mahboob Ur Rahman*, Sahar Imtiaz*, James Gross*,

*School of Electrical Engineering and the ACCESS Linnaeus Center, Royal Institute of Technology (KTH)

Abstract—We study here the problem of Antenna Domain Formation (ADF) in cloud RAN systems, whereby multiple remote radio-heads (RRHs) are each to be assigned to a set of antenna domains (ADs), such that the total interference between the ADs is minimized. We formulate the corresponding optimization problem, by introducing the concept of *interference coupling coefficients* among pairs of radio-heads. We then propose a low-overhead algorithm that allows the problem to be solved in a distributed fashion, among the aggregation nodes (ANs), and establish basic convergence results. Moreover, we also propose a simple relaxation to the problem, thus enabling us to characterize its maximum performance. We follow a layered coordination structure: after the ADs are formed, radio-heads are clustered to perform coordinated beamforming using the well known Weighted-MMSE algorithm. Finally, our simulations show that using the proposed ADF mechanism would significantly increase the sum-rate of the system (with respect to random assignment of radio-heads).

Index Terms—5G, Cloud RAN, radio head assignment, antenna domain formation, interference coupling, block coordination descent

I. INTRODUCTION

The *Cloud-Radio Access Network (C-RAN)* is identified as one of the promising architectures to address the challenges of 5G systems, namely, the requirement for high spectral efficiency within a particularly dense deployment (both users and access nodes) [1]. C-RAN systems are characterized as a centralized solution for interference coordination: remote radio-heads (RRH) act as access nodes, and their baseband processing capabilities vary from full digital signal processing capability (i.e., base stations), to 'dumb antennas' with no baseband capabilities (such as distributed MIMO systems). Such radio-heads are connected via high-capacity (possibly wireless) links to so-called *aggregation nodes (ANs)*, each essentially acting as a large processing unit. Thus, such architectures are natural candidates for interference coordination.

In dense deployments, coordination among base stations was identified as the key to achieving high spectral efficiency: indeed the ideas of Coordinated Multi-point (CoMP) [2], [3] and Interference Alignment (IA) [4], [5] were central to achieve higher spectral efficiency. However, when applied to conventional cellular systems, such techniques have the stringent requirement that they need to be distributed, i.e., to only use local CSI at each node: the overhead associated with such techniques has been identified as a (potentially) limiting factor of the sum-rate gains brought about by techniques such as IA [6], [7], [8] and [9]. This essentially puts hard limits on the effectiveness of the latter techniques. However, this limitation is lifted in the C-RAN architecture, since ANs can be assumed to have perfect CSI of a large area, and can perform coordination in a centralized manner. Earlier related work has been reported in [10] where the authors investigate the beamforming design problem (for sum-rate maximization), in the context of cellular systems with limited backhaul capacity.

In [11], the authors study the problem of dynamic clustering in dense deployments (for joint transmission), by characterizing the statistics of the instantaneous signal-to-interference-and-noise ratio (SINR), via tools from stochastic geometry. In our earlier paper [12], we investigated radio-head coordination (namely coordinated beamforming), in a typical Cloud RAN setup, with a large number of radio-heads and users, served by one AN. In this work, however, we look higher into the coordination hierarchy, by investigating the so-called *Antenna Domain Formation (ADF)* problem: given a set of radio-heads (each serving a set of users), and a set of ANs, what is the best assignment of radio-heads to ANs, such that the total interference leakage between the ADs, is minimized. Studying the latter setup is the main contribution and novelty of this paper. In that sense, we formulate the ADF problem as integer programming problem, and devise an iterative algorithm for solving it. We also relax the latter problem to obtain a lower bound on the maximum performance of our algorithm. Moreover, we investigate the effect of using a layered coordination structure, whereby further coordination mechanism (coordinated beamforming) are put in place. We underline the that fact that this work is currently being extended to a journal form[13].

In the following, we use bold upper-case letters to denote matrices, bold lower-case to denote vectors, and calligraphic letters to denote sets. Furthermore, for a given matrix $\boldsymbol{A}$, $[\boldsymbol{A}]_{i:j}$ denotes the matrix formed by taking columns i to j, of $\boldsymbol{A}$, $\|\boldsymbol{A}\|_F^2$ its Frobenius norm, $|\boldsymbol{A}|$ its determinant, $\boldsymbol{A}^T$ its transpose, and $\boldsymbol{A}^\dagger$ its conjugate transpose . $[\boldsymbol{A}]_{i,j} = a_{i,j}$ denotes element (i,j) in a matrix $\boldsymbol{A}$, and $[\boldsymbol{a}]_i$ element i in a vector $\boldsymbol{a}$. While $\boldsymbol{I}_n$ denotes the $n \times n$ identity matrix, $\boldsymbol{1}_n$ denotes the $n \times 1$ vector of ones, $\mathcal{B}_N$ denotes space of N-dimensional binary vectors, and $\Pi_{\mathcal{S}}[\boldsymbol{x}]$ is the Euclidean projection of a vector $\boldsymbol{x}$, into some (possibly non-convex) set $\mathcal{S}$.
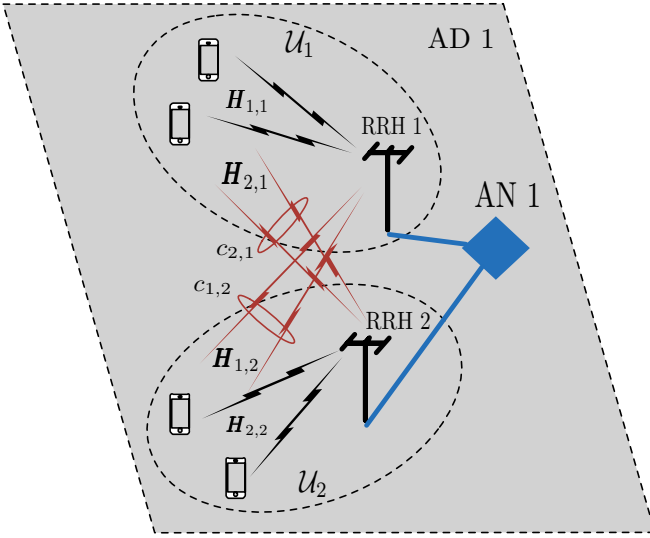
Fig. 1: Toy Example

II. SYSTEM MODEL

Given a large area of interest, i.e. a spatial area of a certain size, with N of RRHs and A ANs, we wish to address the issue of ADF, i.e. which radio heads should be assigned to which AN. In other words, given a set $\mathcal{R}$ of radio heads in an area, the problem is to assign them to a set of ADs (where each AD is controlled by one AN) $\mathcal{A}$, such that total interference coupling between the ADs is minimal. Let $\mathcal{A}$ denote a set of ANs (where $A \triangleq |\mathcal{A}|$), and $\mathcal{R}$ the set of radio-heads where $N = |\mathcal{R}|$.

We assume that radio head $i \in \mathcal{R}$ is equipped with M_i antennas, and serving users which are single-antenna receivers (we assume single antenna users for simplicity of exposition, though this can be extended to multi-antenna receivers). Let $\mathcal{U}_i$ be the set of users served by radio head $i \in \mathcal{R}$. Then, $\boldsymbol{H}_{i,j} \in \mathbb{C}^{|\mathcal{U}_j| \times M_i}$ denotes the channel from the antennas of radio head $i \in \mathcal{R}$, to the users served by radio head $j \in \mathcal{R}$, i.e. $\mathcal{U}_j$, and $\boldsymbol{W}_i \in \mathbb{C}^{M_i \times |\mathcal{U}_i|}$ be the precoder that RRH i employs to serve its users $\mathcal{U}_i$. A small toy example is illustrated in Fig. 1.

This work essentially addresses the interference between ADs, as this type of interference limits significantly the performance of the entire system. It is clear that for any form of operation within the ADs, as well as for any type of interference mitigation technique applied between the ADs, there are more suitable and less suitable AD choices for the radio heads.

We assume that the load has been pre-allocated among all ANs, implying that the kth AN will have a load of $\gamma_k \geq 0$. A special case of this is when the load is equally distributed across all ANs, i.e., γ_k is the same among all the ANs. Furthermore, each RRH is to be assigned to one AN only. Moreover, it is assumed that each AN has perfect CSI of the users to which it is associated.

Hence, the main idea in order to form ADs is to reduce the interference coupling between the different ADs. Denote by $\alpha_{i,j} \in \mathbb{R}_+$ the interference leakage between radio head $i \in \mathcal{R}$ and radio head $j \in \mathcal{R}$, and note that this denotes the leakage in the direction of j, i.e. $\alpha_{i,j}$ is a measure that represents the interference caused by RRH i on terminals associated to RRH j, if i and j are assigned to be in different ADs. Due to asymmetric channel gains and terminal associations, notice in particular that $\alpha_{i,j} \neq \alpha_{j,i}$.

Thus, we define a coupling coefficient $c_{i,j}$ as the interference coupling coefficient between RRH i and RRH j. The latter can be then viewed as the cost of having both radio heads in the same AD. Thus, one intuitive choice for the latter is to select $c_{i,j}$ as the total interference leakage between the two radio heads, i.e.,

$$c_{i,j} = \alpha_{i,j} + \alpha_{j,i} = \|\boldsymbol{H}_{i,j}\boldsymbol{W}_j\|_F^2 + \|\boldsymbol{H}_{j,i}\boldsymbol{W}_i\|_F^2 \quad (1)$$

However, incorporating the precoders into the coupling coefficients makes the overall system quite challenging since the coupling coefficients have to be updated quite frequently. Thus, we formally define the latter quantity as,

$$c_{i,j} = \begin{cases} \|\boldsymbol{H}_{i,j}\|_F^2 + \|\boldsymbol{H}_{j,i}\|_F^2, \ \forall \ i \neq j \\ 0 \ , \forall i = j \end{cases} \quad (2)$$

Let $\boldsymbol{\Psi}$ be the matrix formed by gathering all those coefficients, i.e. $[\boldsymbol{\Psi}]_{i,j} = c_{i,j}, \ \forall \ (i,j) \in \mathcal{R}^2$, where $\boldsymbol{\Psi} \in \mathbb{R}_+^{N \times N}$ is such that $\boldsymbol{\Psi} = \boldsymbol{\Psi}^T$.

Furthermore, denote by $x_{i,k} \in \{0,1\}$ the binary decision variable if RRH i is assigned to AD k (or not), and denote by $\boldsymbol{x}_k \in \{0,1\}^N$ the assignment vector of AN $k \in \mathcal{A}$. We formulate the ADF problem as the following integer programming problem:

$$(P1) \begin{cases} \min f(\{x_{i,j}\}) = \sum_{k=1}^A \sum_{l \neq k}^A \left(\sum_{i=1}^N \sum_{j \neq i}^N c_{i,j} x_{i,k} x_{j,l} \right) \\ \text{s. t. } \sum_{i=1}^N \beta_{i,k} x_{i,k} = \gamma_k, \ \forall k \in \mathcal{A} \\ \sum_{k=1}^A x_{i,k} \leq 1, \ \forall i \in \mathcal{R} \\ x_{i,k} \in \{0,1\}, \ \forall (i,k) \in \mathcal{A} \times \mathcal{R} \end{cases}$$

Note that one can rewrite the above problem in terms of $\{\boldsymbol{x}_k\}$ as follows,

$$(P2) \begin{cases} \min f(\{\boldsymbol{x}_k\}) = \sum_{k=1}^A \sum_{l \neq k}^A \boldsymbol{x}_k^T \boldsymbol{\Psi} \boldsymbol{x}_l \\ \text{s. t. } \sum_{k=1}^A \boldsymbol{x}_k \leq \boldsymbol{1}_N \\ \boldsymbol{\beta}_k^T \boldsymbol{x}_k = \gamma_k, \ \boldsymbol{x}_k \in \mathcal{B}_N, \ \forall k \in \mathcal{A} \end{cases}$$

where the above inequality holds element-wise. In the above, $\sum_{k=1}^A \boldsymbol{x}_k \leq \boldsymbol{1}_N$ is the *assignment constraint* forcing that each RRH is assigned to at most one AD. Moreover, denoting by $\boldsymbol{\beta}_k \in \mathbb{R}_+^N$ the set of loading factors for AD k (such that $\boldsymbol{1}^T \boldsymbol{\beta}_k = 1, \ \forall k \in \mathcal{A}$), $\boldsymbol{\beta}_k^T \boldsymbol{x}_k = \gamma_k$ is the *loading constraint* for the kth AD (i.e. the load profile for AD k). Using this equivalent form, it becomes evident that f indeed is not jointly convex in all the variables, due to the coupling among $\boldsymbol{x}_k$ and $\boldsymbol{x}_l$. However, this does reveal an inherent bi-linear structure of f (taken separately in each variable, f is linear) that we exploit for the optimization. Intuitively, f models the sum-total coupling that exists *between* the individual ADs.

III. PROPOSED ALGORITHM

A. Block Coordinate Descent

The coupling among the variables in $f(\boldsymbol{x}_1, ..., \boldsymbol{x}_A)$, makes $(P2)$ a perfect candidate for a Block-Coordinate Descent (BCD) approach (also known as the Gauss-Siedel method). In a nutshell, BCD works iteratively, by fixing all variables but one block, i.e. fix $(\boldsymbol{x}_1, ..., \boldsymbol{x}_{k-1}, \boldsymbol{x}_{k+1}, ..., \boldsymbol{x}_A)$, and optimize for $\boldsymbol{x}_k$, iteratively. Let n denote the iteration number, i.e., $\boldsymbol{x}_k^{(n)}$ denotes the value of $\boldsymbol{x}_k$ at the nth iteration. At the nth iteration, given that $(\boldsymbol{x}_1^{(n)}, ..., \boldsymbol{x}_{k-1}^{(n)}, \boldsymbol{x}_{k+1}^{(n)}, ..., \boldsymbol{x}_A^{(n)})$ are fixed, $\boldsymbol{x}_k$ is optimized accordingly.

$$\{\boldsymbol{x}_k^{(n+1)}\} = \underset{\boldsymbol{x}_k}{\arg\min}\ f(\boldsymbol{x}_1^{(n+1)}, ..., \boldsymbol{x}_{k-1}^{(n+1)}, \boldsymbol{x}_k, \boldsymbol{x}_{k+1}^{(n)}, ..., \boldsymbol{x}_A^{(n)}),$$
$$= \underset{\boldsymbol{x}_k}{\arg\min}\ f(\boldsymbol{x}_k, \boldsymbol{z}_k^{(n)}), \ \forall k \in \mathcal{A} \qquad (3)$$

where

$$\boldsymbol{z}_k^{(n)} = (\boldsymbol{x}_1^{(n+1)}, ..., \boldsymbol{x}_{k-1}^{(n+1)}, \boldsymbol{x}_{k+1}^{(n)}, ..., \boldsymbol{x}_A^{(n)})$$

denotes the block of fixed variables in the above BCD iteration. Stated differently, in the above, $\boldsymbol{x}_{k-1}^{(n+1)}$ indicates that block $\boldsymbol{x}_{k-1}$ (for instance) has been already updated, while block $\boldsymbol{x}_{k+1}$ hasn't. Thus, as seen from (3), BCD generates a sequence of iterations $\{\boldsymbol{x}_k^{(n)}\}$ that converge to a limit point (this will be formalized later in this section). Moreover, we note that $f(\boldsymbol{x}_k, \boldsymbol{z}_k^{(n)})$ (which denotes the function $f(\boldsymbol{x}_k)$, when the variables in block $\boldsymbol{z}_k^{(n)}$ are fixed), can be rewritten as,

$$f(\boldsymbol{x}_k, \boldsymbol{z}_k^{(n)}) = \boldsymbol{x}_k^T \boldsymbol{\Psi} \left(\sum_{l=1}^{k-1} \boldsymbol{x}_l^{(n+1)} + \sum_{l=k+1}^{A} \boldsymbol{x}_l^{(n)} \right) \triangleq \boldsymbol{x}_k^T \boldsymbol{r}_k^{(n)}$$

$$\text{where } \boldsymbol{r}_k^{(n)} \triangleq \boldsymbol{\Psi} \left(\sum_{l=1}^{k-1} \boldsymbol{x}_l^{(n+1)} + \sum_{l=k+1}^{A} \boldsymbol{x}_l^{(n)} \right) \qquad (4)$$

is referred to as the residual of AD k, at the nth iteration. Moreover, we clearly see that $f(\boldsymbol{x}_k, \boldsymbol{z}_k^{(n)})$ is linear, implying that when all but one block are fixed, f is linear. This indeed shows that $(P2)$ is equivalent to a series of problems that are solved iteratively, and block-by-block. Now that we have described the BCD framework, we focus on the solution of the optimization problem in (3), within each BCD iteration.

With that in mind, the update for $\boldsymbol{x}_k$, i.e. $\boldsymbol{x}_k^{(n+1)}$ in (3), is,

$$\boldsymbol{x}_k^{(n+1)} = \begin{cases} \underset{\boldsymbol{x}_k}{\arg\min}\ f(\boldsymbol{x}_k, \boldsymbol{z}_k^{(n)}) \\ \text{s. t. } \boldsymbol{\beta}_k^T \boldsymbol{x}_k = \gamma_k, \quad \boldsymbol{x}_k \le \boldsymbol{\omega}_k^{(n)}, \quad \boldsymbol{x}_k \in \mathcal{B}_N \end{cases} \qquad (5)$$

where $\boldsymbol{\omega}_k^{(n)}$ is the vector of *residual assignments* dependent on the the assignments of all other ADs.

$$\boldsymbol{\omega}_k^{(n)} \triangleq \mathbf{1}_N - \left(\sum_{l=1}^{k-1} \boldsymbol{x}_l^{(n+1)} + \sum_{l=k+1}^{A} \boldsymbol{x}_l^{(n)} \right) \qquad (6)$$

The above problem is known as a mixed integer linear program, and is NP-hard due to its inherent combinatorial nature. However, there exists many polynomial-time approximation algorithms that, for all practical purposes, solve it globally (such efficient solvers are found in MATLAB and CVX).

It becomes clear at this stage that BCD transforms $(P2)$ into a series of A parallel subproblems, where each can be solved in a distributed way, i.e. locally at each AN.

B. Algorithm Description

The use of BCD for solving $(P2)$ goes hand in hand with making the problem naturally decoupled: when $\{\boldsymbol{x}_l\}_{l \ne k}$ are fixed, the cost function decouples in $\boldsymbol{x}_k$ can thus be solved separately by AD k, without any loss in optimality. In that sense, the optimal update for $\boldsymbol{x}_k$ at AD k, depends on the assignments at all the other ADs, that have to be shared. Given assignments from other ADs, $(\boldsymbol{x}_1^{(n+1)}, ..., \boldsymbol{x}_{k-1}^{(n+1)}, \boldsymbol{x}_{k+1}^{(n)}, ..., \boldsymbol{x}_A^{(n)})$, AD k forms the residual $\boldsymbol{r}_k^{(n)}$, and can proceed to solve its optimization problem locally, and update $\boldsymbol{x}_k^{(n+1)}$. The process is formalized in Algorithm 1.

Algorithm 1 ADF via BCD

Input: $\boldsymbol{\Psi}$, N, A
for $n = 0, 1, \cdots, L-1$ **do**
    // *procedure at each AN*
    obtain $(\boldsymbol{x}_1^{(n+1)}, ..., \boldsymbol{x}_{k-1}^{(n+1)}, \boldsymbol{x}_{k+1}^{(n)}, ..., \boldsymbol{x}_A^{(n)})$ at AD k
    compute residual $\boldsymbol{r}_k^{(n)}$, using (4)
    compute residual assignment $\boldsymbol{\omega}_k$, using (6)
    compute $\boldsymbol{x}_k^{(n+1)}$ as solution to (5)
end for
Output: $\{\boldsymbol{x}_1^{(L)}, ..., \boldsymbol{x}_A^{(L)}\}$

C. Convergence

Let $\{\boldsymbol{x}_k^{(n)}\}$ be the sequence iterates produced by the BCD in (3), and $\{\boldsymbol{x}_k^\star\} \triangleq \lim_{n \to \infty} \{\boldsymbol{x}_k^{(n)}\}$. The monotonic nature of the BCD iterations is established below.

Lemma 1 (Monotonicity). *With each update $\boldsymbol{x}_k^{(n)} \to \boldsymbol{x}_{k+1}^{(n)}$, f is non-increasing. Moreover, the sequence of function iterates $\{f(\boldsymbol{x}_1^{(n)}, ..., \boldsymbol{x}_A^{(n)})\}_n$ converges to a limit point $f(\{\boldsymbol{x}_k^\star\})$.*

    Proof: Refer to Appendix A ∎

Although the above result establishes the convergence of the proposed BCD method, it only establishes convergence to a limit.

Remark 1. Although the updates generated by the BCD iteration are shown to converge monotonically to a limit point, it cannot be established that the latter corresponds to a stationary point of f, namely due to

- the presence of the binary constraint, that prevents the use of advanced BCD convergence results such as [14]
- the coupling in the assignment constraint, i.e. $\boldsymbol{x}_1 + ... + \boldsymbol{x}_A \le \mathbf{1}_N$, cannot be handled by standard BCD convergence results.

D. Performance bounds

Here we attempt to shed light on the maximum performance that Algorithm 1 can deliver. This is achieved by relaxing the original problem in $(P2)$. One of the most well known relaxations for problems such as $(P2)$ is done by relaxing the

binary constraint on $\boldsymbol{x}_k$. In this case however, we note that the loading constraint, $\boldsymbol{\beta}_k^T \boldsymbol{x}_k = \gamma_k$, does not make much sense: a quick look at this case reveals that the loading constraint makes some problems infeasible (this is expected since it is only effective when $\boldsymbol{x}$ is binary). We thus conclude that a sensible relaxation has to involve both the binary constraint, and the loading constraint. The resulting problem becomes,

$$(P3) \begin{cases} \min f(\{\boldsymbol{w}_k\}) = \sum_{k=1}^A \sum_{l \neq k}^A \boldsymbol{w}_k^T \boldsymbol{\Psi} \boldsymbol{w}_l \\ \text{s. t. } \sum_{k=1}^A \boldsymbol{w}_k \leq \mathbf{1}_N, \\ \qquad \boldsymbol{w}_k \in [0,\ 1]^N, \quad \forall k \in \mathcal{A} \end{cases}$$

where $\{\boldsymbol{w}_k\}$, the optimization variables are no longer binary. The fact that the optimal solution of $(P3)$ is a lower bound on the original problem $(P2)$, follows immediately from the relaxation arguments. Despite its simple form, globally solving the above problem is not straightforward, namely due to coupling among the variables, $\{\boldsymbol{w}_k\}$, and that f is not convex in $\{\boldsymbol{w}_k\}$ (since $\boldsymbol{\Psi}$ is not positive-definite). However, we recall that the same BCD procedure that was used to solve $(P2)$, can be applied to the relaxed problem. Thus, the sequence of iterates generated by the BCD is given by,

$$\boldsymbol{w}_k^{(n+1)} = \begin{cases} \underset{\boldsymbol{w}_k}{\text{argmin }} f(\boldsymbol{w}_k, \boldsymbol{z}_k^{(n)}) \\ \text{s. t. } \boldsymbol{w}_k \leq \boldsymbol{\omega}_k^{(n)}, \quad \boldsymbol{w}_k \in [0,\ 1]^N \end{cases} \quad (7)$$

where $f(\boldsymbol{w}_k, \boldsymbol{z}_k^{(n)})$ and $\boldsymbol{\omega}_k^{(n)}$ are given as follows,

$$f(\boldsymbol{w}_k, \boldsymbol{z}_k^{(n)}) = \boldsymbol{w}_k^T \boldsymbol{\Psi} \left(\sum_{l=1}^{k-1} \boldsymbol{w}_l^{(n+1)} + \sum_{l=k+1}^A \boldsymbol{w}_l^{(n)} \right) \triangleq \boldsymbol{w}_k^T \boldsymbol{r}_k^{(n)}$$

$$\boldsymbol{\omega}_k^{(n)} = \mathbf{1}_N - \left(\sum_{l=1}^{k-1} \boldsymbol{w}_l^{(n+1)} + \sum_{l=k+1}^A \boldsymbol{w}_l^{(n)} \right) \quad (8)$$

Let $\{\boldsymbol{w}_k^{(n)}\}$ be the sequence iterates produced by the BCD in (7), and $\{\boldsymbol{w}_k^\star\} \triangleq \lim_{n \to \infty} \{\boldsymbol{w}_k^{(n)}\}$. Then, by applying BCD to relaxed problem $(P3)$, yields the desired lower bound on $(P2)$. Moreover, the resulting solution to $(P3)$, $\{\boldsymbol{w}_k^\star\}$ is such that $f(\{\boldsymbol{x}_k^\star\}) \geq f(\{\boldsymbol{w}_k^\star\})$. We recall at this stage that $\{\boldsymbol{w}_k^\star\}$ doesn't necessarily correspond to an assignment variable. Thus, a natural question is whether one can obtain an estimate of the solution to $(P2)$, $\{\hat{\boldsymbol{x}}_k^\star\}$, from the solution of the relaxed problem $\{\boldsymbol{w}_k^\star\}$. The relation between $\hat{\boldsymbol{x}}_k^\star$ and $\boldsymbol{w}_k^\star$ can be formalized as follows,

$$\hat{\boldsymbol{x}}_k^\star = \Pi_{\mathcal{D}_k}[\boldsymbol{w}_k^\star],\ \forall k \in \mathcal{A}$$
$$\text{where } \mathcal{D}_k = \{\boldsymbol{w} \mid \boldsymbol{w} \in \mathcal{B}_N,\ \boldsymbol{\beta}_k^T \boldsymbol{w} = \gamma_k\} \quad (9)$$

Stated differently, $\hat{\boldsymbol{x}}_k^\star$ is the Euclidean projection of $\boldsymbol{w}_k^\star$, on the non-convex set $\mathcal{D}_k$. However, a closer look at the latter reveals that the projection in this case, does not always yield a unique point $\hat{\boldsymbol{x}}_k^\star$. Consequently, although the solution of the relaxed problem, $\boldsymbol{w}_k^\star$, cannot be used as a basis for assignment, we indeed use it as a lower bound on the cost function value that Algorithm 1 yields.

E. System-level Operation

We use Algorithm 1 to perform the assignment of RRHs to ADs. Then, there are additional coordination mechanisms deployed within each AD: cooperation within the latter is done via clusters of cooperating RRHs, that in turn are formed based on geographical distance (using algorithms such as the K-means). Moreover, all the radio heads in the latter cluster perform coordinated beamforming (CB) (using the well known Weighted MMSE (WMMSE) [15]), to iteratively optimize their precoders. The steps in this coordination hierarchy are detailed in the table below.

System level operation:
- Assign users to RRHs based on highest channel energy
- Compute coupling coefficients matrix $\boldsymbol{\Psi}$
- Run ADF algorithm to obtain RRH-to-AD assignment
for each AD do
  - Cluster RRH into C clusters using k-means (optional)
  - Run WMMSE (within each cluster) to optimize precoders
end for

The result of combining the latter two 'coordination mechanisms' is a hierarchical cooperation model: at the highest level, RRHs are assigned into ADs (using Algorithm 1), then, RRHs are grouped to form coordination clusters (where each is performing coordinated beamforming). The resting network structure is illustrated in Fig. 2. Next, we mathematically formalized the operation and performance of the system.

1) Sum-rate performance: We assume for simplicity that each RRH is serving J users, and that the size of each cluster, $|\mathcal{C}_m^{(l)}|$, is the same. Moreover, let C denote the number of clusters within each AD. Let $\mathcal{C}_m^{(l)}$ denote the mth cluster ($1 \leq m \leq C$) in AN l ($1 \leq l \leq A$): then $\mathcal{C}_m^{(l)} \subset \mathcal{R}$ is the set of cooperating radio heads within the lth AD (Fig 2).

For shorthand notation, we denote by i_k, the ith user ($1 \leq i \leq J$), served by RRH k ($1 \leq k \leq K$). Then, $\boldsymbol{h}_{i_k,j}^{(l,m)}$ is the (MISO) channel from RRH j, to user i_k, in $\mathcal{C}_m^{(l)}$. Similarly, we define $\boldsymbol{v}_{i_k}^{(l,m)}$ at the transmit precoder, used to serve user i_k, in $\mathcal{C}_m^{(l)}$. Letting P_t denote the transmit power of all radio-heads, the SINR of user i_k, in $\mathcal{C}_m^{(l)}$, is given by (10). Thus, treating interference as noise at the users, the *achievable sum-rate* of the system is as follows,

$$R_\Sigma = \sum_{l \in \mathcal{A}} \sum_{m=1}^C \sum_{i=1}^J \sum_{k=1}^K \log_2(1 + \text{SINR}_{i_k}^{(l,m)}) \quad (11)$$

Under this setup we advocate, each user is subject to residual interference from users within its coordination cluster, interference emanating from users in other clusters (but still within the same AD), as well as interference coming from all users present in other ADs.

2) Practical Aspects: The matrix of coupling coefficients $\boldsymbol{\Psi}$ should available at all the ANs, prior to the start of the algorithm: the latter can be "populated" sequentially, by having each AD estimate all channels (both to its served users, and to users from other ADs) via training, in an orthogonal manner.

Due to the fact that the assignment variables are binary, one only needs a low-rate control link between all the ANs. In the
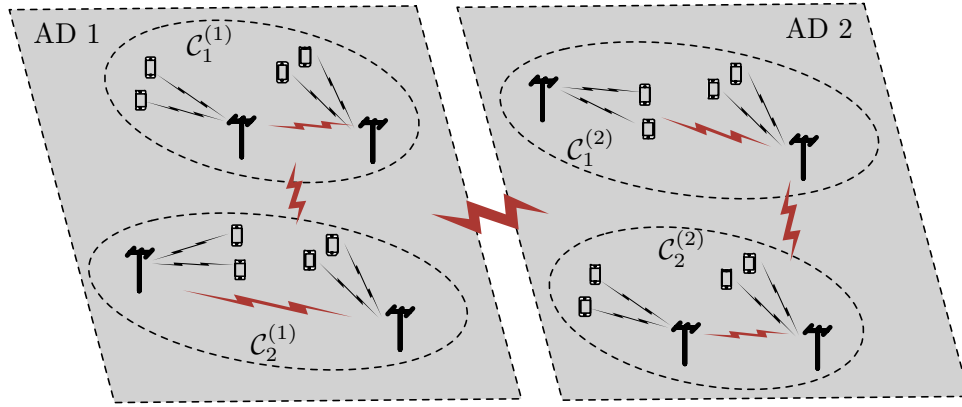
Fig. 2: System-level Operation

$$\text{SINR}_{i_k}^{(l,m)} = \frac{P_t |(\boldsymbol{h}_{i_k,i}^{(l,m)})^\dagger \boldsymbol{v}_{i_k}^{(l,m)}|^2}{\sum\limits_{(p,q)\neq(i,k)} P_t |(\boldsymbol{h}_{p_q,i}^{(l,m)})^\dagger \boldsymbol{v}_{i_k}^{(l,m)}|^2 + \sum\limits_{(r,s)\neq(l,m)} \sum\limits_{(p,q)} P_t |(\boldsymbol{h}_{p_q,i}^{(r,s)})^\dagger \boldsymbol{v}_{p_q}^{(r,s)}|^2 + (\sigma_{i_k}^{(l,m)})^2} \tag{10}$$

case where $A = 2$, due to the bilinear structure of f, only one iteration of Algorithm 1 is required for convergence: this is quite beneficial since it keeps the communication overhead to a minimal level. Moreover, one can initialize the algorithm with several feasible solutions, and pick the best optimal solution among them.

IV. NUMERICAL RESULTS

A. Simulation Setup

Both radio heads and users are dropped uniformly within the area of interest (their positions are kept fixed throughout the simulation), where no mobility is considered. Then, for each simulation run, channels are generated randomly: all channels are complex i.i.d, assumed to be slowly block fading.

We first investigate a system where $N = 16$ radio heads are deployed, each equipped with $M = 4$ transmit antennas and serving $J = 2$ single-antenna users (for a total of $K = 32$ users). The RRHs are to be assigned to one of two ANs (i.e., $A = 2$). In our simulations we assume that within each AD, radio heads form one cluster (i.e. $C = 1$), thereby forming *global coordination (GC)* cluster within each AD. Then, WMMSE is employed to iteratively optimize the precoders within each cluster. Moreover, we assume that the loading factors are identical, whereby the RRH load is split equally among the two ADs, i.e. $\beta_k = 1$, $\forall k$ and $\gamma_k = N/A$, $\forall k$.

For the assignment of radio heads to ANs, we benchmark our proposed scheme, Algorithm 1, against a *randomized assignment* where radio heads are randomly assigned to each of the ANs. Despite the fact that the latter is not a good choice, it is however intended to be used as a *lower bound on the performance* of our algorithm (the latter assignment still takes into account the equal-loading constraint, and the fact that each RRH is assigned to one AD only). Moreover, for this particular (relatively small) case, we are able to find the globally optimal solution to the ADF problem, via *exhaustive* search of $(P1)$.

We also investigate two (extreme) special cases of ADF,

- *ADF based on instantaneous CSI* is the case detailed in (2) where matrix of coupling coefficients $\boldsymbol{\Psi}$ is based on instantaneous CSI: in this case, $\boldsymbol{\Psi}$ is updated every time the channel changes, and consequently, the ADs have to be recomputed at every channel realization
- *ADF based on statistical CSI*, where the coupling coefficients are given by,

$$c_{i,j} = \begin{cases} \|\tilde{\boldsymbol{H}}_{i,j}\|_F^2 + \|\tilde{\boldsymbol{H}}_{j,i}\|_F^2, \ \forall \ i \neq j \\ 0 \ , \forall i = j \end{cases} \tag{12}$$

where, analogously to $\boldsymbol{H}_{i,j}$, we define $\tilde{\boldsymbol{H}}_{i,j} \in \mathbb{C}^{|\mathcal{U}_j| \times M_i}$ as the matrix of *pathloss factors* from the antennas of radio head $i \in \mathcal{R}$, to the users of radio head $j \in \mathcal{R}$. Unlike the instantaneous CSI case, here $\boldsymbol{\Psi}$ is computed at the beginning of the simulation, the ADF performed, and remains static throughout the simulation run (since users are static).

B. Results

The sum-rate performance (refer to (11)) that results from the above setup is shown in Fig 3, where we compare the performance of our algorithm, against both the random assignment and the optimal exhaustive search. Instantaneous CSI is assumed for all schemes. We can clearly see a significant performance gap between the performance of our scheme, and that of the benchmark. Moreover, looking at the performance of the exhaustive search, reveals that there are rather large gains from finding the globally optimal solution to the ADF problem.

We next investigate the extent to which replacing Instantaneous CSI with long-term Statistical CSI, degrades the sum-rate performance of the system in question (we follow the exact same setup used for the instantaneous CSI case). As shown in Fig 4, we see the same trends discussed just above.
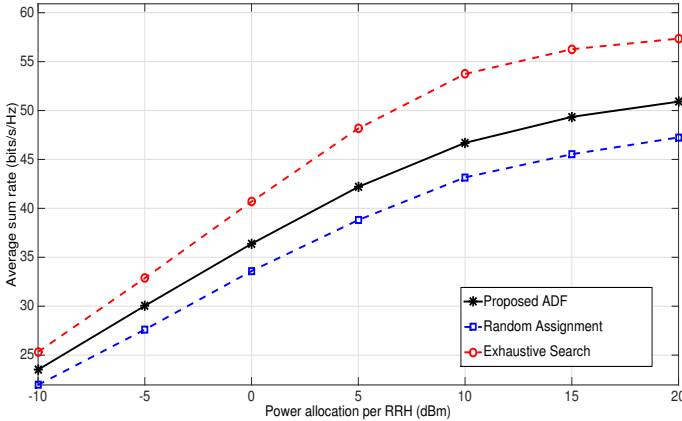
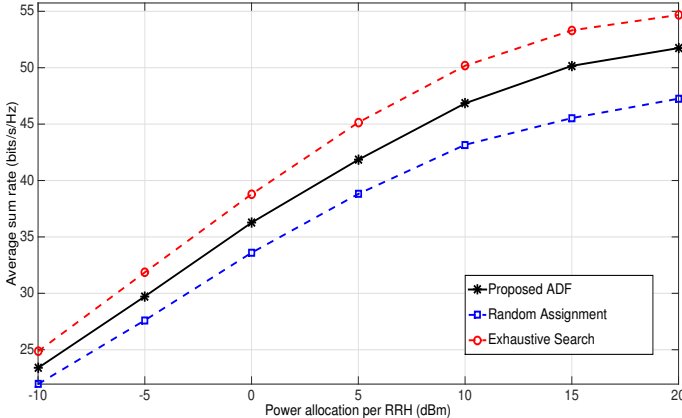Fig. 3: Average sum-rate performance using instantaneous CSI $M = 4, N = 16, K = 32, C = 1, J = 2$

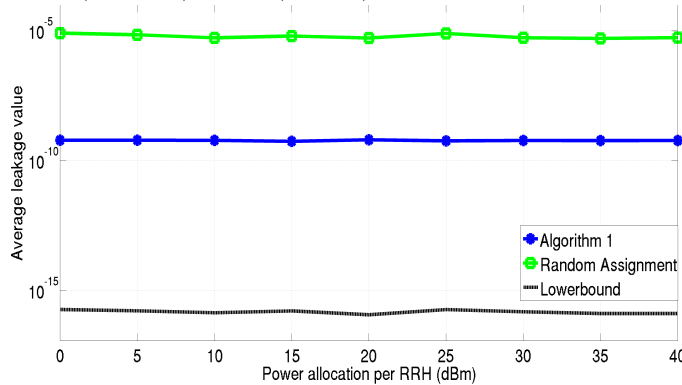Fig. 4: Average sum-rate performance using statistical CSI $M = 4, N = 16, K = 32, C = 1, J = 2$

Fig. 5: Average leakage value ($M = 4, N = 16, K = 32, C = 1, J = 2$)

More importantly, as far as our algorithm is concerned, the degradation observed by going from instantaneous to statistical CSI, is extremely negligible. This can be exploited to massively reduce the overhead and complexity of the system (detailed in next subsection).

Following the same setup as above, we shed light on the behavior of the proposed ADF algorithm 1, as a function of the SNR. In Fig 5 we show the average value of f that is achieved by Algorithm 1, as well as the lower bound discussed in III-D. Though our algorithm greatly outperforms the benchmark, it is relatively far from the lower-bound (derived from the relaxed problem $(P3)$): however, we reiterate the fact that solutions such as the latter, do not correspond to feasible solutions for the ADF problem (thus are potentially not achievable within our setup).

C. Discussions

The numerical results all point to a good performance brought about by the application of our proposed method (both in terms of achievable sum-rates for the system, and total interference leakage between the ADs). Moreover, as shown in Algorithm 1, the communication overhead requirement among the ANs is quite negligible since it consists of binary vectors only. With that in mind, the overhead would consist of obtaining all the CSI needed to compute $\boldsymbol{\Psi}$, and propagate it to all the ANs (as often as needed). Though this might seem too high, our simulations also clearly indicated that for relatively static settings, ADF based on pathloss (i.e., location-based) has virtually the same performance as the one based on instantaneous CSI.

V. CONCLUSION AND FUTURE WORK

In this work we tackled the problem of ADF in C-RAN systems by formulating it as an integer programming problem. We employ the well-known BCD framework for solving the problem and devising an iterative algorithm for that purpose. We shed light on the convergence of the algorithm, as well as its 'maximal performance' via simple relaxations. Our simulations reveal that this approach promises to deliver good sum-rate performance, in typical C-RAN deployments.

As mentioned earlier, we started to investigate more analytical aspects of the ADF problem, namely, better relaxation techniques (e.g. Lagrange Relaxation), decomposition techniques (e.g. Dantzig-Wolfe Decomposition), and dual problem analysis. The latter will provide us with insights into fundamental lower bounds for the ADF problem. In addition, we extended the problem formulation used here, to include both the channel and precoder effect, rather than just the channel energy (as seen in (1)). All the above issues are investigated in great detail, in our subsequent work [13]. We also wish to investigate more practical scenarios, namely the so-called hybrid CSI case, consisting of a mix between instantaneous and statistical CSI.

VI. ACKNOWLEDGMENT

APPENDIX

A. Proof of Lemma 1

Note that the following is a direct consequence of (5)

$$f(\{\boldsymbol{x}_k^{(n)}\}) \geq f(\boldsymbol{x}_1^{(n+1)}, \boldsymbol{z}_1^{(n)}) \geq f(\boldsymbol{x}_2^{(n+1)}, \boldsymbol{z}_2^{(n)})...$$
$$\geq f(\boldsymbol{x}_A^{(n+1)}, \boldsymbol{z}_A^{(n)}) \triangleq f(\{\boldsymbol{x}_k^{(n+1)}\})$$

where the last equality follows from the fact that $f(\boldsymbol{x}_A^{(n+1)}, \boldsymbol{z}_A^{(n)})$ corresponds to the case where all variables $(\boldsymbol{x}_1, ..., \boldsymbol{x}_A)$, are updated. It follows that the sequence $\{f(\boldsymbol{x}_1^{(n)}, ..., \boldsymbol{x}_A^{(n)})\}_n$ converges monotonically to a limit point f_0

REFERENCES

[1] M. D6.2, "Initial report on horizontal topics, first results and 5G system concept," March 2014.

[2] D. Gesbert, S. Hanly, H. Huang, S. Shamai Shitz, O. Simeone, and W. Yu, "Multi-cell MIMO cooperative networks: A new look at interference," *Selected Areas in Communications, IEEE Journal on*, vol. 28, no. 9, pp. 1380–1408, 2010.

[3] E. Bjornson, N. Jalden, M. Bengtsson, and B. Ottersten, "Optimality properties, distributed strategies, and measurement-based evaluation of coordinated multicell ofdma transmission," *Signal Processing, IEEE Transactions on*, vol. 59, pp. 6086–6101, Dec 2011.

[4] V. Cadambe and S. Jafar, "Interference alignment and degrees of freedom of the K -user interference channel," *Information Theory, IEEE Transactions on*, vol. 54, no. 8, pp. 3425–3441, 2008.

[5] M. Maddah-Ali, A. Motahari, and A. Khandani, "Communication over MIMO X channels: Interference alignment, decomposition, and performance analysis," *IEEE Transactions on Information Theory*, vol. 54, pp. 3457 –3470, Aug. 2008.

[6] O. El Ayach, A. Lozano, and R. Heath, "On the overhead of interference alignment: Training, feedback, and cooperation," *Wireless Communications, IEEE Transactions on*, vol. 11, no. 11, pp. 4192–4203, 2012.

[7] O. El Ayach, A. Lozano, and R. Heath, "Optimizing training and feedback for MIMO interference alignment," in *Signals, Systems and Computers (ASILOMAR), 2011 Conference Record of the Forty Fifth Asilomar Conference on*, pp. 1717–1721, 2011.

[8] S. Peters and R. Heath, "User partitioning for less overhead in MIMO interference channels," *IEEE Transactions on Wireless Communications*, vol. 11, pp. 592 –603, Feb. 2012.

[9] A. Lozano, R. Heath, and J. Andrews, "Fundamental limits of cooperation," *Information Theory, IEEE Transactions on*, vol. 59, pp. 5213–5226, Sept 2013.

[10] B. Dai and W. Yu, "Sparse beamforming design for network mimo system with per-base-station backhaul constraints," in *Signal Processing Advances in Wireless Communications (SPAWC), 2014 IEEE 15th International Workshop on*, pp. 294–298, June 2014.

[11] N. Lee, R. Heath, D. Morales-Jimenez, and A. Lozano, "Base station cooperation with dynamic clustering in super-dense cloud-RAN," in *Globecom Workshops (GC Wkshps), 2013 IEEE*, pp. 784–788, Dec 2013.

[12] M. Rahman, H. Ghauch, S. Imtiaz, and J. Gross, "RRH clustering and transmit precoding for interference-limited 5G CRAN downlink," *preprint available at http://arxiv.org/abs/1510.07672*, 2015.

[13] H. Ghauch, M. Rahman, S. Imtiaz, and J. Gross, "Antenna domain formation: Performance limits and practical algorithms," *manuscript in preparation*, 2016.

[14] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *Journal of Optimization Theory and Applications*, vol. 109, no. 3, pp. 475–494, 2001.

[15] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, "An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4331–4340, 2011.