Distributed Cooperative Beamforming in Multi-Source Multi-Destination Clustered Systems

Nikolaos Chatzipanagiotis, *Student Member, IEEE,* Yupeng Liu, *Student Member, IEEE,* Athina Petropulu, *Fellow, IEEE* and Michael M. Zavlanos, *Member, IEEE*

***Abstract*—We consider the scenario of a multi-cluster network, in which each cluster contains multiple single-antenna source destination pairs that communicate simultaneously over the same channel. The communications are supported by cooperating amplify-and-forward relays, which perform beamforming. While the communications take place within the cluster, there is inter-cluster as well as intra-cluster interference. The beamforming weights are obtained so that the total relay transmit power is minimized, while a certain signal-to-interference-plus-noise-ratio (SINR) at the destinations is met. First, we show that a computationally efficient approximate solution is attainable by relaxing the original NP-hard non-convex problem to a semidefinite optimization form. Then, we propose a decentralized method to solve the convex problem, based on the recently developed Accelerated Distributed Augmented Lagrangians (ADAL) algorithm, a distributed optimization technique that achieves fast convergence rates. Our decentralized solution allows for each cluster to compute its own beamforming weights, while coordinating with other clusters via appropriate message exchanges. Two different approaches are presented, differing in the message exchange patterns between clusters. The performance of the decentralized scheme is demonstrated via simulations.**

***Index Terms*—Cooperative beamforming, multi-cluster systems, multi-source multi-destination systems, multiuser peer-to-peer relay networks, distributed optimization, augmented Lagrangian.**

I. INTRODUCTION

Cooperative (or relaying) approaches for wireless communications have the potential for significant performance improvement, such as extended coverage of the network, throughput enhancement and energy savings [1]–[21]. In cooperative beamforming, a set of relays form a "virtual antenna array" and retransmit weighted versions of the source signals (decode-and-forward (DF) relaying), or weighted versions of the received signals (amplify-and-forward (AF) relaying). By exploiting constructive interference effects, the relays focus the transmitted power on the destinations' locations, thus increasing the directional channel gain. By achieving spatial multiplexing, cooperative beamforming can support the communications of multiple, distinct, single-antenna, source-destination pairs that overlap both in time and frequency. This scenario is also referred to as multiuser peer-to-peer relay networks [5]–[16].

In general, the per-node throughput capacity of a wireless ad-hoc network reduces rapidly as the network size increases [22]. Therefore, it is often preferable to divide the network nodes into multiple clusters, with each cluster containing nodes which have distinct sub-goals, or are geographically close to each other, e.g., applications involving networks of mobile wireless robots [23, 24].

In this paper we consider a multi-cluster network, in which multiuser peer-to-peer relay communications take place inside each cluster, while the intra-cluster communications cause inter-cluster interference. In this context, the relay weights are computed based on channel second-order statistics, so that the total relay transmit power is minimized, while meeting certain signal-to-interference-plus-noise-ratio (SINR) constraints at the destinations. First, we show that a computationally efficient approximate solution is attainable by relaxing the original NP-hard non-convex problem employing semidefinite relaxation (SDR) techniques [25]–[28]. Second, we propose a distributed approach to solve the relaxed problem, which allows for each cluster to compute its optimal beamforming weights based on information exchanges with neighboring clusters only. Such a distributed approach obviates the need for a a central processing unit that has access to the channel statistics of all clusters and obtains the relay weights; centralized approaches do not scale well with the number of network nodes, resulting in high complexity and long delays. Our proposed distributed approach is based on *Accelerated Distributed Augmented Lagrangians* (ADAL) [29]. ADAL is a distributed optimization method that relies on *augmented Lagrangians* (AL), a regularization technique that is obtained by adding a quadratic penalty term to the ordinary Lagrangian of a problem [30]. Compared to standard distributed optimization techniques, such as *dual decomposition* [31], AL methods converge very fast and do not require strict convexity of the objective function [30, 31]. The latter is a necessary feature for our multi-cluster relay beamforming problem, since the objective function under consideration is affine. At the same time, it was shown in [29] that, for a number of different applications, ADAL exhibits a significant improvement in convergence speed compared to existing AL techniques, such as the ADMM [32] and the DQA [33].

We propose two different ways to apply ADAL to the multi-cluster beamforming problem, termed *Direct* and *Indirect*,

This work is supported by NSF CNS Grants #1239339 and #1239188.

Nikolaos Chatzipanagiotis and Michael M. Zavlanos are with the Dept. of Mechanical Engineering and Materials Science, Duke University, Durham, NC, 27708, USA, {n.chatzip,michael.zavlanos}@duke.edu. Yupeng Liu is with Alcatel-Lucent, New Providence, NJ, 07974, USA, yupeng.liu@alcatel-lucent.com. Athina Petropulu is with the Dept. of Electrical and Computer Engineering, Rutgers, the State University of New Jersey, Piscataway, NJ, 08854, USA, athinap@rutgers.edu.

that allow us to model different message exchange patterns (necessary for the iterative execution of ADAL) between the individual clusters. Specifically, the message exchange pattern in the Direct method is determined by the coupling SINR constraints due to inter-cluster interference. On the other hand, the message exchanges in the Indirect method can be defined arbitrarily by the user. Both approaches rely on transforming the SINR coupling constraints to a linear form by introducing appropriate auxiliary variables. We show, via numerical experiments, that the Direct method is generally more efficient than the Indirect. However, the flexibility of the Indirect method in selecting the message exchange pattern between clusters might make it more appropriate for certain applications.

To the best of our knowledge, there is no prior work showing that the multi-cluster relay beamforming problem is amenable to a decentralized solution. The closest scenarios considered in the literature are those of multi-cell downlink beamfroming [34, 35], which do not involve relays and thus the formulation is considerably simpler; the two AF communication stages of the relay problem that we consider in this paper give rise to several additional interference terms that have to be taken into account. The beamforming weight design in [34] and [35] employs respectively the dual decomposition method and the ADMM. Although one could use similar methods as in [34, 35] to solve our problem, the ADAL method converges faster according to the simulation results presented in Section IV.

The rest of the paper is organized as follows: In Section II, we first discuss the single cluster relay beamforming scenario. Then, we formulate the multi-cluster problem and propose to pose it as a convex optimization problem using SDR. In Section III, we present two different ways to obtain a decentralized solution to the convex multi-cluster problem by applying ADAL. Finally, in Section IV, we present simulation results to verify the validity of our approach.

II. Relay Beamforming

To facilitate understanding of the multi-cluster scenario, we first formulate the cooperative beamforming problem for a single cluster. The solution for this problem can be found in [14]. Then, in Section II-B we formulate and solve the multi-cluster problem.

A. Single Cluster case

In the single cluster scenario, the goal is to allow communication of multiple single-antenna source-destination pairs, which transmit simultaneously using the same channel. The transmission takes place in two stages, i.e., two consecutive time-slots. In the first stage, all sources transmit, while in the second stage the relays retransmit the signals that they received in an AF fashion. A simple case with two source-destination pairs and three relays is depicted in Fig. 1.

Consider a network composed of an index set $\mathcal{M} = \{1,\ldots,M\}$ of sources $S_m, \forall m \in \mathcal{M}$, users (destinations) $U_m, \forall m \in \mathcal{M}$ and a corresponding set $\mathcal{L} = \{1,\ldots,L\}$ of dedicated, single-antenna relay nodes $R_l, \forall l \in \mathcal{L}$. Source S_m

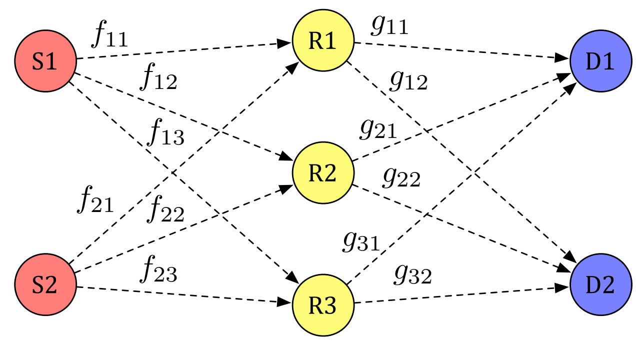


Fig. 1. Simultaneous communication between 2 sources and 2 destinations with the help of 3 relays. The signal transmitted from source 1 (S1) is intended for destination 1 (D1), while the signal transmitted from source 2 (S2) is intended for destination 2 (D2). Signals from S1 and S2 that reach D2 and D1, respectively, are considered interference. Also shown are the channel gains f_{ij} between sources and relays, and g_{ij} between relays and destinations.

wishes to communicate with user U_m. During the first communication stage, every source S_m transmits the signal $\sqrt{P_0}s_m$, where P_0 is the common power, and $s_m \in \mathbb{C},\ m = 1,\ldots,M$ denote the information symbols, which are independent identically distributed (i.i.d.) with unit power. The received signal at every relay R_l is given by

$$x_l = \sqrt{P_0}\sum_{m=1}^{M} f_{ml}s_m + v_l,$$

where $\mathbb{C}$ denotes the set of complex numbers, $f_{ml} \in \mathbb{C}$ is the channel between source S_m and relay R_l, and $v_l \in \mathbb{C}$ is the noise at relay R_l, assumed to have zero mean and unit variance. The channel coefficients are treated as random, i.i.d., independent between different paths. This assumption is valid when the nodes are sufficiently separated. It is also assumed that the channel coefficients are independent of the source signals and the noise. Correspondingly, the received signal vector at all relays can be expressed in matrix form as

$$\mathbf{x} = \sqrt{P_0}\mathbf{F}\mathbf{s} + \mathbf{v},$$

where $\mathbf{s} = [s_1,\ldots,s_M]^T \in \mathbb{R}^M$, $\mathbf{x} = [x_1,\ldots,x_L]^T \in \mathbb{C}^L$, $\mathbf{v} = [v_1,\ldots,v_L]^T \in \mathbb{C}^L$, with $(\cdot)^T$ denoting the transposition operation, and

$$\mathbf{F} = \begin{bmatrix} f_{11} & \ldots & f_{M1} \\ \vdots & \ddots & \vdots \\ f_{1L} & \ldots & f_{ML} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 & \ldots & \mathbf{f}_M \end{bmatrix} \in \mathbb{C}^{L\times M}$$

is a channel state matrix, with $\mathbf{f}_m = [f_{m1},\ldots,f_{mL}]^T \in \mathbb{C}^L$ denoting the channel gain column vector from source S_m to all relays.

During the second communication stage, the relays retransmit, in an AF fashion, a linear transformation of their received signals. We can express this linear transformation as a multiplication of $\mathbf{x}$ with a beamforming matrix $\mathbf{W} \in \mathbb{C}^{L\times L}$. Hence, the vector $\mathbf{t} \in \mathbb{C}^L$ of relay transmissions can be written as

$$\mathbf{t} = \mathbf{W}\mathbf{x} = \sqrt{P_0}\mathbf{W}\mathbf{F}\mathbf{s} + \mathbf{W}\mathbf{v},$$

Since the relays are physically separated, they do not have access to the signals received at other relays. Each relay operates only on its own received signal, and thus the beamforming matrix is diagonal, i.e. $\mathbf{W} = \text{diag}\{w_1, \ldots, w_L\}$, where w_l denotes the complex weight with which relay R_l multiplies its received signal.

By similar reasoning as above, the received signal vector $\mathbf{y} \in \mathbb{C}^M$ at the destinations equals

$$\mathbf{y} = \mathbf{G}\mathbf{t} + \mathbf{z} = \sqrt{P_0}\mathbf{G}\mathbf{W}\mathbf{F}\mathbf{s} + \mathbf{G}\mathbf{W}\mathbf{v} + \mathbf{z},$$

where $\mathbf{z} = [z_1, \ldots, z_M]^T \in \mathbb{C}^M$ denotes the vector stacking i.i.d random noise components with zero mean and unit variance, and

$$\mathbf{G} = \begin{bmatrix} g_{11} & \cdots & g_{L1} \\ \vdots & \ddots & \vdots \\ g_{1M} & \cdots & g_{LM} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_1 & \ldots & \mathbf{g}_M \end{bmatrix}^T \in \mathbb{C}^{M \times L},$$

is a channel state matrix, with $\mathbf{g}_m = [g_{1m}, \ldots, g_{Lm}]^T \in \mathbb{C}^L$ denoting the channel gain column vector from all relays to user $U_m,\ m \in \mathcal{M}$. The received signal at user U_m can be divided in three components that capture: i) the desired signal originating from source S_m, ii) interference due to sources other than S_m that constitute interference, and iii) noise at the user, i.e.,

$$\begin{aligned} y_m = &\ \mathbf{g}_m^T\mathbf{t} + z_m \qquad (1) \\ = &\ \underbrace{\mathbf{g}_m^T\mathbf{W}\mathbf{f}_m s_m}_{\text{Desired}} + \underbrace{\sum_{j \in \mathcal{M}}^{j \neq m} \mathbf{g}_m^T\mathbf{W}\mathbf{f}_j s_j + \mathbf{g}_m^T\mathbf{W}\mathbf{v}}_{\text{Interference}} + \underbrace{z_m}_{\text{Noise}}, \end{aligned}$$

A reasonable optimality criterion for determining the beamforming weights is the minimization of the total relay transmit power, P_T, subject to satisfying certain Quality of Service requirements at all destinations, namely enforcing user-specific SINR bounds $\gamma_m > 0$. Since the channels are in general random, we will use the average transmit power. Therefore, we can pose the cooperative beamforming problem as the following optimization problem:

$$\begin{array}{ll} \min_{\mathbf{W}} & P_T(\mathbf{W}) \\ \text{s.t.} & \text{SINR}_m(\mathbf{W}) \geq \gamma_m, \ \ \forall m = 1, \ldots, M. \end{array} \qquad (2)$$

The total average transmit power at the relays equals

$$\begin{aligned} P_T(\mathbf{W}) = &\ \mathbb{E}\{\|\mathbf{t}\|_F^2\} \\ = &\ \mathbb{E}\left\{ \text{Tr}\left[\left(\sqrt{P_0}\mathbf{W}\mathbf{F}\mathbf{s} + \mathbf{W}\mathbf{v} \right) \left(\sqrt{P_0}\mathbf{W}\mathbf{F}\mathbf{s} + \mathbf{W}\mathbf{v} \right)^{\mathcal{H}} \right] \right\} \\ = &\ \text{Tr}\Big(P_0\mathbf{W}\mathbb{E}\{\mathbf{F}\mathbf{F}^{\mathcal{H}}\}\mathbf{W}^{\mathcal{H}} \Big) + \text{Tr}\Big(\mathbf{W}\mathbf{W}^{\mathcal{H}} \Big), \end{aligned}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. The expectation in the first equation is taken over source signals and channels, while in the second and third equations, due to the i.i.d. and unit power assumption on the s_m's, the signal terms have been already averaged out and the expectation refers to the channels. Due to our assumptions, $\mathbb{E}\{\mathbf{F}\mathbf{F}^{\mathcal{H}}\}$ is a diagonal matrix. Since $\mathbf{W}$ is also a diagonal matrix, we can use this property to express the sum transmit power as

$$P_T = \mathbf{w}^{\mathcal{H}}\mathbf{R}_T\mathbf{w},$$

where $\mathbf{w} = \begin{bmatrix} w_1, \ldots, w_L \end{bmatrix}^T \in \mathbb{C}^L$ is a column vector containing all the diagonal elements of $\mathbf{W}$, and

$$\mathbf{R}_T = P_0 \text{diag}\Big\{ \sum_{m \in \mathcal{M}} \mathbb{E}\{|f_{m1}|^2\}, \ldots, \sum_{m \in \mathcal{M}} \mathbb{E}\{|f_{mL}|^2\} \Big\} + \mathbf{I}_L,$$

where $|\cdot|$ denotes the magnitude of a complex number. Based on (1), the SINR at every user U_m is defined as

$$\text{SINR}_m \triangleq \frac{\mathbb{E}\Big(\overbrace{P_0|\mathbf{g}_m^T\mathbf{W}\mathbf{f}_m s_m|^2}^{\text{Desired}} \Big)}{\mathbb{E}\Big(\underbrace{P_0 \sum_{j \in \mathcal{M}}^{j \neq m} |\mathbf{g}_m^T\mathbf{W}\mathbf{f}_j s_j|^2}_{\text{Interference}} + \underbrace{||\mathbf{g}_m^T\mathbf{W}\mathbf{v}||^2 + |z_m|^2}_{\text{Noise}} \Big)},$$

where the term $P_0 \sum_{j \in \mathcal{M}}^{j \neq m} |\mathbf{g}_m^T\mathbf{W}\mathbf{f}_j s_j|^2$ represents interference at user U_m caused by signals intended for other users, the term $||\mathbf{g}_m^T\mathbf{W}\mathbf{v}||^2$ denotes noise at the relays that was propagated to the user, and $|z_m|^2$ denotes noise at the user level. The expectation in the above equation refers to everything that is random, i.e., signals, channels, noise. Observe that the average SINR is defined as the ratio of the expected values, which is different than the expected value of the ratio. This definition is frequently used in communications textbooks, e.g. [36], and in published works related to the problem considered here [4, 6, 7, 14, 20].

Similar as before, we can manipulate the SINR expression to write it in a more compact matrix form

$$\text{SINR}_m = \frac{\overbrace{P_0\mathbf{w}^{\mathcal{H}}\mathbf{R}_S^m\mathbf{w}}^{\text{Desired}}}{\underbrace{P_0\mathbf{w}^{\mathcal{H}}\mathbf{R}_I^m\mathbf{w}}_{\text{Interference}} + \underbrace{\mathbf{w}^{\mathcal{H}}\mathbf{R}_v^m\mathbf{w} + 1}_{\text{Noise}}}.$$

The desired signal matrix for user U_m is Hermitian

$$\mathbf{R}_S^m = \mathbb{E}\{(\mathbf{f}_m^T \odot \mathbf{g}_m^T)^{\mathcal{H}}(\mathbf{f}_m^T \odot \mathbf{g}_m^T)\},$$

with $\odot$ denoting the Hadamard (entry-wise) product. The corresponding interference matrix is also Hermitian

$$\mathbf{R}_I^m = \sum_{j \in \mathcal{M}}^{j \neq m} \mathbb{E}\{(\mathbf{f}_j^T \odot \mathbf{g}_m^T)^{\mathcal{H}}(\mathbf{f}_j^T \odot \mathbf{g}_m^T)\},$$

and the respective noise matrix is diagonal

$$\mathbf{R}_v^m = \text{diag}\big\{\mathbb{E}\{|g_{1m}|^2\}, \ldots, \mathbb{E}\{|g_{Lm}|^2\}\big\}.$$

Utilizing the above notation, the single-cluster optimization problem (2) can be compactly written as

$$\begin{aligned} \min_{\mathbf{w}} \quad & \mathbf{w}^{\mathcal{H}}\mathbf{R}_T\mathbf{w} \\ \text{s.t.} \quad & \mathbf{w}^{\mathcal{H}}\mathbf{Q}^m\mathbf{w} \geq 1, \ \ \forall m = 1, \ldots, M, \end{aligned} \tag{3}$$

where $\mathbf{Q}^m = \frac{P_0}{\gamma_m}\mathbf{R}_S^m - P_0\mathbf{R}_I^m - \mathbf{R}_v^m$.

B. Multiple Clusters case

In this section we consider a multi-cluster network, in which, neighboring clusters' communications interfere. A simple setup with two clusters is depicted in Fig. 2. In this case, the beamforming decisions of the relays of each cluster must also take into account the interference caused to and from the other clusters' operation. This introduces three new terms in the SINR of each user. First, there is a term quantifying the interference on the relays of each cluster, exerted by the transmissions of other clusters' sources during the first communication stage, which then propagates to the users of this cluster after the second stage transmission. Second, there is also interference on the users of each cluster exerted by the signals transmitted from the relays of other clusters that are intended for other users. Finally, the noise at the relays of all clusters propagates to the users of each cluster after the second stage transmissions.

Define a set $\mathcal{N} = \{1, \ldots, N\}$ of clusters, where each cluster $C_n, \forall n \in \mathcal{N}$ is now composed of a set $\mathcal{M}_n = \{1, \ldots, M\}$ of single antenna source-destination pairs, and a set $\mathcal{L}_n = \{1, \ldots, L\}$ of dedicated relays. We denote the m-th user (destination) of the n-th cluster as $U_{nm}, \ \forall n \in \mathcal{N}, m \in \mathcal{M}_n$, the respective source as S_{nm}, and the relays as $R_{nl}, \ \forall n \in \mathcal{N}, l \in \mathcal{L}_n$. Note that we assume for simplicity of notation, and without loss of generality, that all clusters contain the same number of source destination pairs M and relays L.

In the multi-cluster scenario, the received signal at every relay R_{nl} is a superposition of signals originating from the sources of all clusters

$$x_{nl} = \sqrt{P_0}\sum_{j\in\mathcal{N}}\sum_{m\in\mathcal{M}_j} f_{jm,nl}s_{jm} + v_{nl},$$

where, again, P_0 is the common transmit power of all sources and $s_{nm} \in \mathbb{C}$ denotes the, normalized to unit power, information symbol transmitted by source S_{nm}. Also, $v_{nl} \sim \mathcal{CN}(0, 1)$ is the noise at relay R_{nl} and $f_{jm,nl}$ denotes the channel gain between source S_{jm} and relay R_{nl}. Re-writing in matrix form, the received signal vector at all relays of cluster C_n is

$$\mathbf{x}_n = \sum\nolimits_{j\in\mathcal{N}} \sqrt{P_0}\mathbf{F}_{jn}\mathbf{s}_j + \mathbf{v}_n,$$

where $\mathbf{s}_j = [s_{j1}, \ldots, s_{jM}]^T \in \mathbb{C}^M$, $\mathbf{x}_n = [x_{n1}, \ldots, x_{nL}]^T \in \mathbb{C}^L$, $\mathbf{v}_n = [v_{n1}, \ldots, v_{nL}]^T \in \mathbb{C}^L$. The matrix $\mathbf{F}_{jn} \in \mathbb{C}^{L\times M}$ is defined as the channel state matrix containing the channels from all sources of cluster C_j to all the relays of cluster C_n, i.e.,

$$\mathbf{F}_{jn} = \begin{bmatrix} f_{j1,n1} & \cdots & f_{jM,n1} \\ \vdots & \ddots & \vdots \\ f_{j1,nL} & \cdots & f_{jM,nL} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_{j1,n} & \cdots & \mathbf{f}_{jM,n} \end{bmatrix},$$

where $\mathbf{f}_{jm,n} = [f_{jm,n1}, \ldots, f_{jm,nL}]^T \in \mathbb{C}^L$ denotes the channel gain vector from source S_{jm} to all relays of cluster C_n.

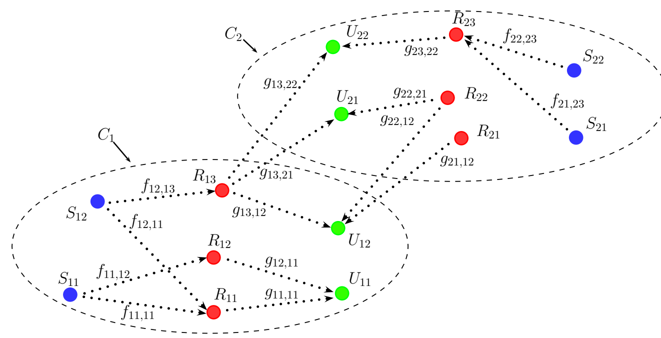


Fig. 2. A $N = 2$ multi-cluster relay beamforming scenario, with $M = 2$ source-destination pairs (green and blue dots respectively) and $L = 3$ dedicated relays (red dots) for each cluster. Note that only a portion of all the available channels is drawn in order to avoid congestion.

Similar to the single cluster scenario, during the second communication stage the relays of cluster C_n retransmit, in an AF fashion, a linear transformation of their respective received signals $\mathbf{x}_n$, i.e.,

$$\mathbf{t}_n = \mathbf{W}_n\mathbf{x}_n = \sqrt{P_0}\mathbf{W}_n\big(\sum\nolimits_{j\in\mathcal{N}}\mathbf{F}_{jn}\mathbf{s}_j\big) + \mathbf{W}_n\mathbf{v}_n,$$

where $\mathbf{t}_n \in \mathbb{C}^L$ denotes the forwarded signal vector and $\mathbf{W}_n \in \mathbb{C}^{L\times L}$ is the corresponding beamforming matrix of cluster C_n. Recall that we consider the case where every relay node carries a single antenna, which translates into the beamforming matrix being diagonal, i.e. $\mathbf{W}_n = \text{diag}\{w_{n1}, \ldots, w_{nL}\} \in \mathbb{C}^{L\times L}$, where w_{nl} denotes the complex weight with which relay R_{nl} multiplies its received signal.

The received signal vector $\mathbf{y}_n \in \mathbb{C}^M$ for all users of each cluster C_n is now a superposition of signals from the relays of all clusters, and can be expressed as

$$\begin{aligned} \mathbf{y}_n &= \sum_{j\in\mathcal{N}}\mathbf{G}_{jn}\mathbf{t}_j + \mathbf{z}_n \\ &= \sum_{j\in\mathcal{N}}\Big(\sqrt{P_0}\mathbf{G}_{jn}\mathbf{W}_j\big(\sum_{i\in\mathcal{N}}\mathbf{F}_{ij}\mathbf{s}_i\big) + \mathbf{G}_{jn}\mathbf{W}_j\mathbf{v}_j\Big) + \mathbf{z}_n, \end{aligned} \tag{4}$$

where $\mathbf{z}_n = [z_{n1}, \ldots, z_{nM}]^T \in \mathbb{C}^M$ denotes the vector of i.i.d random noise components $z_{nm} \sim \mathcal{CN}(0, 1)$ at user U_{nm}. The matrix $\mathbf{G}_{jn} \in \mathbb{C}^{M\times L}$ is defined as the channel state matrix containing the channels from all relays of C_j to all the users of C_n, i.e.,

$$\mathbf{G}_{jn} = \begin{bmatrix} g_{j1,n1} & \cdots & g_{jL,n1} \\ \vdots & \ddots & \vdots \\ g_{j1,nM} & \cdots & g_{jL,nM} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{j,n1} & \cdots & \mathbf{g}_{j,nM} \end{bmatrix}^T,$$

with $\mathbf{g}_{j,nm} = [g_{j1,nm}, \ldots, g_{jL,nm}]^T \in \mathbb{C}^L$ denoting the channel gain column vector from all relays of C_j to U_{nm}. The received signal at user U_{nm} is given by the m-th entry of the vector $\mathbf{y}_n$ in (4). The pertinent expression is

$$
\begin{aligned}
y_{nm} =\ & \sum\nolimits_{j\in\mathcal{N}} \mathbf{g}_{j,nm}^T \mathbf{t}_j + z_{nm} \\
=\ & \underbrace{\sqrt{P_0}\mathbf{g}_{n,nm}^T \mathbf{W}_n \mathbf{f}_{nm,n} s_{nm}}_{\text{Desired}} \\
& + \underbrace{\mathbf{g}_{n,nm}^T \mathbf{W}_n \Big(\sum\nolimits_{i\in\mathcal{M}_n}^{i\neq m} \sqrt{P_0}\mathbf{f}_{ni,n} s_{ni}\Big)}_{\text{Intra-Cluster Interference from same cluster's sources other than } S_{nm}} \\
& + \underbrace{\mathbf{g}_{n,nm}^T \mathbf{W}_n \Big(\sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \sqrt{P_0}\mathbf{F}_{jn} \mathbf{s}_j\Big)}_{\text{Inter/Intra-Cluster Interference from out-of-cluster sources}} \\
& + \underbrace{\sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \mathbf{g}_{j,nm}^T \mathbf{W}_j \Big(\sum\nolimits_{i\in\mathcal{N}} \sqrt{P_0}\mathbf{F}_{ij} s_i\Big)}_{\text{Inter-Cluster Interference}}, \\
& + \underbrace{\sum\nolimits_{j\in\mathcal{N}} \mathbf{g}_{j,nm}^T \mathbf{W}_j \mathbf{v}_j + z_{nm}}_{\text{Noise}},
\end{aligned}
$$

where we can now see more clearly how the three aforementioned, additional interference terms that arise in the multi-cluster scenario affect the formulation. Specifically, the term labeled *"Inter/Intra-Cluster Interference from out-of-cluster sources"* denotes the interference on the relays of each cluster, exerted by the transmissions of other clusters' sources during the first communication stage. This interference propagates to the users in the second stage when the relays of the same cluster transmit, hence the hybrid characterization *Inter/Intra-Cluster Interference*. Also, the term labeled *"Inter-Cluster Interference"* denotes the interference on the users of each cluster, exerted by signals transmitted by the relays of other clusters that are intended for other users. Finally, notice that the term labeled *Noise* now also includes the noise at the relays of all clusters that propagates to the users of each cluster after the second stage transmissions. We assume that the two stages of the AF protocol are synchronized for the whole network, such that interference from source transmissions affecting user receptions directly is not possible.

Subsequently, the average SINR of user U_{nm} is defined as

$$
\begin{aligned}
\text{SINR}_{nm} \triangleq\ & \mathbb{E}\left(\underbrace{P_0|\mathbf{g}_{n,nm}^T \mathbf{W}_n \mathbf{f}_{nm,n} s_{nm}|^2}_{\text{Desired}}\right) \Big/ \\
& \mathbb{E}\left(\underbrace{P_0 \sum_{i\in\mathcal{M}_n}^{i\neq m} |\mathbf{g}_{n,nm}^T \mathbf{W}_n \mathbf{f}_{ni,n} s_{ni}|^2}_{\text{Intra-cluster interference}} + \underbrace{\sum_{j\in\mathcal{N}} |\mathbf{g}_{j,nm}^T \mathbf{W}_j \mathbf{v}_j|^2}_{\text{Noise}}\right. \\
& + \underbrace{P_0 \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \sum\nolimits_{k\in\mathcal{M}_j} |\mathbf{g}_{n,nm}^T \mathbf{W}_n \mathbf{f}_{jk,n} \mathbf{s}_{jk}|^2}_{\text{Inter/Intra-cluster interference}} \\
& \left. + \underbrace{P_0 \sum_{j\in\mathcal{N}}^{j\neq n} \sum_{i\in\mathcal{N}} \sum_{k\in\mathcal{M}_i} |\mathbf{g}_{j,nm}^T \mathbf{W}_j \mathbf{f}_{ik,j} \mathbf{s}_{ik}|^2}_{\text{Inter-cluster interference}} + \underbrace{|z_{nm}|^2}_{\text{Noise}}\right).
\end{aligned}
$$

Thus, the multi-cluster beamforming problem entails finding $\mathbf{W}_n$ that solves the optimization problem

$$
\begin{aligned}
\min_{\{\mathbf{W}_n, \forall n\in\mathcal{N}\}} \quad & \sum\nolimits_{n\in\mathcal{N}} P_T^n(\mathbf{W}_n) \qquad (5) \\
\text{s.t.} \quad & \text{SINR}_{nm}(\mathbf{W}_n) \geq \gamma_{nm}, \ \ \forall n \in \mathcal{N}, \ m \in \mathcal{M}_n,
\end{aligned}
$$

where the average, total transmited power at the relays of cluster C_n is calculated as

$$
\begin{aligned}
P_T^n \ =\ & \mathbb{E}\{\|\mathbf{t}_n\|_F^2\} \\
=\ & \sum_{j\in\mathcal{N}} \text{Tr}\big(P_0 \mathbf{W}_n \mathbb{E}\{\mathbf{F}_{jn}\mathbf{F}_{jn}^{\mathcal{H}}\} \mathbf{W}_n^{\mathcal{H}}\big) \ + \ \text{Tr}\big(\mathbf{W_n}\mathbf{W_n^{\mathcal{H}}}\big).
\end{aligned}
$$

To facilitate further exposition, and, also, to accomodate for the manipulations in Section III, in what follows we will express (5) in a matrix form. To this end, the total transmited power at the relays of C_n can be written as

$$
P_T^n = \mathbf{w}_n^{\mathcal{H}} \mathbf{R}_T^n \mathbf{w}_n,
$$

where $\mathbf{w}_n = \big[w_{n1}, \ldots, w_{nL}\big]^T \in \mathbb{C}^L$ is a column vector containing all the diagonal elements of $\mathbf{W}_n$, and

$$
\begin{aligned}
\mathbf{R}_T^n = \mathbf{I}_L + P_0 \sum\nolimits_{j\in\mathcal{N}} \sum\nolimits_{m\in\mathcal{M}_j} \text{diag}\Big\{ & \mathbb{E}\{|f_{jm,n1}|^2\}, \ldots, \\
& , \ldots, \mathbb{E}\{|f_{jm,nL}|^2\}\Big\},
\end{aligned}
$$

with $\mathbf{I}_L$ denoting the identity matrix of size L. Doing the same for the SINR expression, we define for every $n \in \mathcal{N}, m \in \mathcal{M}_n$ the desired signal matrices as

$$
\mathbf{R}_S^{nm} = \mathbb{E}\{(\mathbf{f}_{nm,n}^T \odot \mathbf{g}_{n,nm}^T)^{\mathcal{H}} (\mathbf{f}_{nm,n}^T \odot \mathbf{g}_{n,nm}^T)\},
$$

the intra-cluster interference matrices as

$$
\mathbf{R}_I^{nm} = \sum\nolimits_{i\in\mathcal{M}_n}^{i\neq m} \mathbb{E}\{(\mathbf{f}_{ni,n}^T \odot \mathbf{g}_{n,nm}^T)^{\mathcal{H}} (\mathbf{f}_{ni,n}^T \odot \mathbf{g}_{n,nm}^T)\},
$$

and the inter/intra-cluster interference matrices as

$$
\begin{aligned}
& \mathbf{R}_{II}^{nm} = \\
& \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \sum\nolimits_{k\in\mathcal{M}_j} \mathbb{E}\{(\mathbf{f}_{jk,n}^T \odot \mathbf{g}_{n,nm}^T)^{\mathcal{H}} (\mathbf{f}_{jk,n}^T \odot \mathbf{g}_{n,nm}^T)\}.
\end{aligned}
$$

Moreover, for every $n \in \mathcal{N}, \ m \in \mathcal{M}_n$ and $j \in \mathcal{N} \backslash \{n\}$ we define the inter-cluster interference matrices as

$$
\mathbf{R}_{IC}^{j,nm} = \sum_{i\in\mathcal{N}} \sum_{k\in\mathcal{M}_i} \mathbb{E}\{(\mathbf{f}_{ik,j}^T \odot \mathbf{g}_{j,nm}^T)^{\mathcal{H}} (\mathbf{f}_{ik,j}^T \odot \mathbf{g}_{j,nm}^T)\},
$$

and, finally, the noise matrices as

$$
\mathbf{R}_v^{j,nm} = \text{diag}\{\mathbb{E}\{|g_{j1,nm}|^2\}, \ldots, \mathbb{E}\{|g_{jL,nm}|^2\}\}.
$$

Note that all the above matrices are Hermitian. Using the above notation, the SINR_{nm} is equivalently expressed as

$$
\begin{aligned}
\text{SINR}_{nm} = \Big(P_0 \mathbf{w}_n^{\mathcal{H}} \mathbf{R}_S^{nm} \mathbf{w}_n\Big) \Big/ \Big(& P_0 \mathbf{w}_n^{\mathcal{H}} \mathbf{R}_I^{nm} \mathbf{w}_n \\
& + \ P_0 \mathbf{w}_n^{\mathcal{H}} \mathbf{R}_{II}^{nm} \mathbf{w}_n \ + \ P_0 \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \mathbf{w}_j^{\mathcal{H}} \mathbf{R}_{IC}^{j,nm} \mathbf{w}_j^{\mathcal{H}} \\
& + \ \sum\nolimits_{i\in\mathcal{N}} \mathbf{w}_i^{\mathcal{H}} \mathbf{R}_v^{i,nm} \mathbf{w}_i + 1\Big).
\end{aligned}
$$

Then, problem (5) can be written in matrix form as

$$\begin{aligned} \min_{\{\mathbf{w}_n, \forall n \in \mathcal{N}\}} \quad & \sum\nolimits_{n\in\mathcal{N}} \mathbf{w}_n^{\mathcal{H}} R_T^n \mathbf{w}_n \\ \text{s.t.} \quad & \mathbf{w}_n^{\mathcal{H}} \mathbf{Q}^{nnm} \mathbf{w}_n + \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \mathbf{w}_j^{\mathcal{H}} \mathbf{Q}^{jnm} \mathbf{w}_j \geq 1, \\ & \forall n \in \mathcal{N},\ m \in \mathcal{M}_n, \end{aligned} \tag{6}$$

where we have further defined the matrices

$$\mathbf{Q}^{nnm} = \frac{P_0}{\gamma_{nm}} \mathbf{R}_S^{nm} - P_0 \mathbf{R}_I^{nm} - P_0 \mathbf{R}_{II}^{nm} - \mathbf{R}_v^{n,nm}$$

$$\text{and} \quad \mathbf{Q}^{jnm} = -P_0 \mathbf{R}_{IC}^{j,nm} - \mathbf{R}_v^{j,nm},$$

to obtain a more compact notation for each SINR constraint. Note that the term $\mathbf{w}_j^{\mathcal{H}} \mathbf{Q}^{jnm} \mathbf{w}_j$ essentially gathers all the terms that depend on the beamforming decisions of cluster C_j and appear in the SINR constraint of user U_{nm}.

Since the matrices $\mathbf{Q}^{jnm}$, $\mathbf{Q}^{nnm}$ will be, in general, indefinite, it follows that the optimization problem (6) belongs in the class of nonconvex Quadratically Constrained Quadratic Programming (QCQP) problems, which are NP-hard to solve. Nevertheless, by defining the variables $\mathbf{X}_j \triangleq \mathbf{w}_j \mathbf{w}_j^{\mathcal{H}},\ \ \forall\, j \in \mathcal{N}$ and using the fact that $\mathbf{w}_j^{\mathcal{H}} \mathbf{Q}^{jnm} \mathbf{w}_j = \mathrm{Tr}(\mathbf{X}_j \mathbf{Q}^{jnm})$, we can express (6) in the equivalent form [25]

$$\begin{aligned} \min_{\{\mathbf{X}_n, \forall n \in \mathcal{N}\}} \quad & \sum\nolimits_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n \mathbf{R}_T^n) \\ \text{s.t.} \quad & \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{nnm}) + \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \mathrm{Tr}(\mathbf{X}_j \mathbf{Q}^{jnm}) \geq 1, \\ & \mathbf{X}_n \in \mathbb{S}_+^L, \quad \forall n \in \mathcal{N},\ m \in \mathcal{M}_n, \\ & \mathrm{rank}(\mathbf{X}_n) = 1, \quad \forall n \in \mathcal{N}. \end{aligned} \tag{7}$$

where $\mathbf{X}_j \in \mathbb{S}_+^L$ imposes the (convex) constraint that matrix $\mathbf{X}_j$ belongs to the cone of symmetric, positive semidefinite matrices of dimension L. Note that, since $\mathbf{Q}^{jnm}$ is Hermitian and $\mathbf{X}_j$ is symmetric, it follows that $\mathrm{Tr}(\mathbf{X}_j \mathbf{Q}^{jnm}) = \mathrm{Tr}(\mathbf{X}_j \mathfrak{Re}(\mathbf{Q}^{jnm}))$ which means that the inequality constraint in (7) is well defined, where $\mathfrak{Re}(\cdot)$ returns the real part of a complex number.

Problem (7) is equivalent to (6) and still nonconvex because of the nonconvex rank constraint. Nevertheless, the rest of the problem is convex, which motivates the relaxation of the rank constraint in order to obtain a problem that is manageable to solve. The resulting SDR of problem (7) becomes

$$\begin{aligned} \min_{\{\mathbf{X}_n, \forall n \in \mathcal{N}\}} \quad & \sum\nolimits_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n \mathbf{R}_T^n) \\ \text{s.t.} \quad & \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{nnm}) + \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \mathrm{Tr}(\mathbf{X}_j \mathbf{Q}^{jnm}) \geq 1, \\ & \mathbf{X}_n \in \mathbb{S}_+^L, \quad \forall n \in \mathcal{N},\ m \in \mathcal{M}_n. \end{aligned} \tag{8}$$

Note that, by dropping the rank constraints, we essentially enlarge the feasible set. Hence, in general, the relaxation (8) will only yield an approximate solution to (7), with an optimal value that provides a lower bound for the original problem. Therefore, the optimizers $\mathbf{X}_j^*,\ \forall j \in \mathcal{N}$ of (8) will not be rank-one in general, due to the relaxation. If they are, then they will be the optimal solution to the original problem (7). If not, randomization techniques [37] can be employed to obtain a rank one matrix.

Remark 1 *Observe that, similar to [4, 14, 15, 20], the above formulation assumes knowledge of the second order statistics of channel state information (CSI). In a practical setting, this can be obtained based on past observations.*

Remark 2 *The inequality constraints in* (8) *must be active at the optimal solution (satisfied as equalities), because if they were not, we would be able to decrease the magnitudes of* $\mathbf{X}_n$ *further, thus invalidating the optimality assumption.*

III. Distributed Relay Beamforming

Since the beamforming decisions in (8) are coupled in the constraint set, a central processing unit would have to be employed to gather the data involving the second order statistics of all channels, compute the optimal solution and then transmit the optimal beamforming weights, expressed in the form of the beamforming matrices $\mathbf{X}_n$, to the corresponding relays. However, this centralized approach would introduce congestion, delays, and would suffer from poor scalability, as the cluster population grows.

In this section we describe a distributed algorithm to solve (8). Our method is distributed in the sense that it allows for each cluster to compute its own optimal beamforming matrix, by performing individual computations based only on locally available information. We utilize the distributed algorithm ADAL [29], which we have recently developed for the solution of convex optimization problems with linear coupling constraints. ADAL is based on the AL framework [30] and eliminates the requirement for strict convexity, which is a necessary condition in simple dual decomposition methods. At the same time, it was shown in [29] that, for a number of different applications, ADAL exhibits a significant improvement in convergence rates compared to existing AL techniques, such as the *Alternating Directions Method of Multipliers* (ADMM) [32] and the *Diagonal Quadratic Approximation* (DQA) [33].

We present two different ways to implement ADAL on our particular multi-cluster beamforming problem, depending on how we express the coupling constraint set of (8). In the rest of this section, we describe these two possible implementations of ADAL, termed *Direct* and *Indirect* for reasons to become transparent later, and discuss their practical applications.

A. Direct method

ADAL is a primal-dual iterative scheme, where each iteration consists of three steps. First, every cluster solves a local convex optimization problem, cf. (14), which, by assuming that only neighboring clusters contribute interference, requires access to the previous primal-dual iterates of neighboring clusters only. Then, every cluster updates and transmits its primal variables according to (16). Finally, after receiving the updated primal variables from neighboring clusters, every cluster updates the dual variables corresponding to its users' constraints according to (17).

To apply ADAL we need to reformulate (8) so that the coupling constraints are affine. For this, define auxiliary variables $\zeta_{njm}, \forall n, j \in \mathcal{N},\ m \in \mathcal{M}_n$ that express the amount of "influence", namely, either the desired signal power or interference exerted by all the relays of cluster C_n on each user U_{jm} of the system. In particular,

$$\begin{aligned} \zeta_{nnm} &= \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{nnm}) - 1 \\ &= \mathrm{Tr}\left(\mathbf{X}_n \big(\frac{P_0}{\gamma_{nm}} \mathbf{R}_S^{nm} - P_0 \mathbf{R}_I^{nm} - \mathbf{R}_v^{n,nm}\big)\right) - 1 \end{aligned} \tag{9}$$

denotes the desired signal power for every user U_{nm} belonging in cluster C_n, while

$$\begin{aligned} \zeta_{njm} &= \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{njm}) \\ &= \mathrm{Tr}\left(\mathbf{X}_n \big(- P_0 \mathbf{R}_{IC}^{j,nm} - \mathbf{R}_v^{j,nm}\big)\right) \end{aligned} \tag{10}$$

denotes the interference exerted by cluster C_n on user U_{jm} that belongs to another cluster C_j, that is for every $j \in \mathcal{N}\backslash\{n\}$ and $m \in \mathcal{M}_j$. Furthermore, define the vector $\boldsymbol{\zeta}_n = [\zeta_{n11}, \ldots, \zeta_{nNM}]^T \in \mathbb{R}^{NM}$ stacking all the "influences" of C_n. Then, problem (8) can be equivalently written as

$$\begin{aligned} \min_{\{\mathbf{X}_n, \forall n \in \mathcal{N}\}} \quad & \sum\nolimits_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n \mathbf{R}_T^n) \\ \text{s.t.} \quad & \sum\nolimits_{n\in\mathcal{N}} \boldsymbol{\zeta}_n = \mathbf{0} \\ & \zeta_{nnm} = \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{nnm}) - 1, \quad \forall n \in \mathcal{N}\ m \in \mathcal{M}_n \\ & \zeta_{njm} = \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{njm}), \forall n \in \mathcal{N}, j \in \mathcal{N}\backslash\{n\}, m \in \mathcal{M}_j \\ & \mathbf{X}_n \in \mathbb{S}_+^L, \quad \forall n \in \mathcal{N}, \end{aligned} \tag{11}$$

where $\mathbf{0}$ is the zero vector of dimension NM. Note that we have replaced the inequality SINR constraints of (8) with the equality constraints $\sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n = \mathbf{0}$ in (11). This is acceptable since the SINR inequality constraints in (8) must be active at the optimal solution, i.e., satisfied as equalities; recall Rem. 2. The idea behind transforming (8) into (11) is that now the problem involves local constraints for each cluster, except for the coupling $\sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n = \mathbf{0}$ which is a simple affine constraint and thus amenable to distributed implementation using the ADAL.

The augmented Lagrangian associated with (11) is

$$\begin{aligned} \Lambda(\mathbf{X}, \boldsymbol{\zeta}, \boldsymbol{\lambda}) = & \underbrace{\sum_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n \mathbf{R}_T^n) + \boldsymbol{\lambda}^T \sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n}_{\text{Ordinary Lagrangian}} \\ & + \underbrace{\frac{\rho}{2} \| \sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n \|_2^2}_{\text{Penalty term}}, \end{aligned} \tag{12}$$

where $\boldsymbol{\lambda} = [\lambda_{11}, \ldots, \lambda_{NM}]^T \in \mathbb{R}^{NM}$ is the vector of Lagrange multipliers (dual variables), $\boldsymbol{X} = \{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N\}$ and $\boldsymbol{\zeta} = \{\boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_N\}$ denote the collection of all primal and auxiliary variables respectively, and $\rho \in \mathbb{R}_+$ is a properly defined penalty coefficient. Note that we include only the constraint $\sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n = 0$ in (12), because the rest of the constraints are local at each cluster C_n. For simplicity of notation, we collectively denote the set of points satisfying these local constraints of each cluster C_n as

$$\begin{aligned} Z_n = \Big\{ & \boldsymbol{\zeta}_n \in \mathbb{R}^{NM} |\ \zeta_{nnm} = \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{nnm}) - 1,\ \forall m \in \mathcal{M}_n, \\ & \zeta_{njm} = \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{njm}),\ \forall j \in \mathcal{N}\backslash\{n\}, m \in \mathcal{M}_j \Big\}. \end{aligned}$$

As already mentioned, the presence of the quadratic penalty term in (12) destroys the separability property of the ordinary Lagrangian. ADAL overcomes this limitation by defining *local augmented Lagrangians* for every cluster C_n

$$\begin{aligned} \Lambda_n(\mathbf{X}_n, \boldsymbol{\zeta}_n, \tilde{\boldsymbol{\zeta}}_j, \boldsymbol{\lambda}) = & \mathrm{Tr}(\mathbf{X}_n \mathbf{R}_T^n) + \boldsymbol{\lambda}^T \boldsymbol{\zeta}_n \\ & + \frac{\rho}{2} \| \boldsymbol{\zeta}_n + \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \tilde{\boldsymbol{\zeta}}_j \|_2^2, \end{aligned} \tag{13}$$

where we introduce variables $\tilde{\boldsymbol{\zeta}}_j$, denoting the primal variables that are controlled by C_j but communicated to C_n for optimization of its local Lagrangian Λ_n. With respect to C_n, these are considered fixed parameters. ADAL is an iterative procedure according to which, at each iteration k, each cluster C_n begins by finding the minimizers $\hat{\boldsymbol{\zeta}}_n^k$ of its local augmented Lagrangian, as

$$\hat{\boldsymbol{\zeta}}_n^k = \underset{\mathbf{X}_n \in \mathbb{S}_+^L, \boldsymbol{\zeta}_n \in Z_n}{\mathrm{argmin}} \ \Lambda_n(\mathbf{X}_n, \boldsymbol{\zeta}_n, \tilde{\boldsymbol{\zeta}}_j^k, \boldsymbol{\lambda}^k). \tag{14}$$

A key observation here is that each cluster C_n does not actually need global information to calculate (14), as it might appear at first sight by looking at the penalty term of each local AL. Although computing the penalty terms appears to require access to all $\tilde{\boldsymbol{\zeta}}_j,\ \forall j \in \mathcal{N}\backslash\{n\}$, one can readily observe that

$$\| \boldsymbol{\zeta}_n + \sum_{j\in\mathcal{N}}^{j\neq n} \tilde{\boldsymbol{\zeta}}_j \|_2^2 = \sum_{i\in\mathcal{N}} \sum_{m\in\mathcal{M}_i} \Big(\zeta_{nim} + \sum_{j\in\mathcal{N}}^{j\neq,n} \tilde{\zeta}_{jim} \Big)^2 \tag{15}$$

where we recall that ζ_{jim} denotes the "influence" that C_j exerts on user U_{im}. In practical applications, each C_n will exert non-negligible interference (above a specified threshold) on a subset $\mathcal{B}_n \subseteq \{U_{11}, \ldots, U_{NM}\}$ of the set of active users and, consequently, we can set to 0 all $\zeta_{nim},\ \forall\, U_{im} \notin \mathcal{B}_n$. Correspondingly, the summation terms in (15) for users $U_{im} \notin \mathcal{B}_n$ that do not experience interference from the operation of C_n are just constant terms in the optimization step (14) and can be neglected. In other words, each C_n only needs information from those clusters that exert non-negligible "influence" on the users belonging in $\mathcal{B}_n$. Therefore, we can formally define the message-exchange neighborhood of cluster C_n as the set of clusters $\mathcal{C}_n = \{C_j :\ j \in \mathcal{N},\ \mathcal{B}_j \bigcap \mathcal{B}_n \neq \varnothing\}$.

After calculating $\hat{\boldsymbol{\zeta}}_n^k$ according to (14), each cluster $C_n,\ \forall n \in \mathcal{N}$ updates its estimates $\tilde{\boldsymbol{\zeta}}_n$ that will be communicated to its neighbors $C_j \in \mathcal{C}_n$ according to

$$\tilde{\boldsymbol{\zeta}}_n^{k+1} = \tilde{\boldsymbol{\zeta}}_n^k + \tau(\hat{\boldsymbol{\zeta}}_n^k - \tilde{\boldsymbol{\zeta}}_n^k), \tag{16}$$

where τ is a stepsize, the determination of which is critical to the convergence properties of the method. Finally, the dual update is performed according to

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \tau\rho \sum\nolimits_{n\in\mathcal{N}} \tilde{\boldsymbol{\zeta}}_n^{k+1}. \tag{17}$$

Algorithm 1 Direct Method

Set $k = 0$ and determine estimates for the sets $\mathcal{C}_n$ and $\mathcal{I}_{nm},\ \forall n \in \mathcal{N},\ m \in \mathcal{M}_n$. Every cluster C_n initializes and transmits its primal $\tilde{\boldsymbol{\zeta}}_n^0$ and dual $\boldsymbol{\lambda}_n^0$ variables.

1. Every cluster C_n minimizes its local AL according to (14), after receiving the primal $\tilde{\boldsymbol{\zeta}}_i^k$ and dual $\boldsymbol{\lambda}_j^k$ variables from clusters $C_j \in \mathcal{C}_n$.
2. Every cluster C_n updates and transmits its primal variables $\tilde{\boldsymbol{\zeta}}_i^{k+1}$ according to (16).
3. Every cluster C_n updates and transmits the dual variables of its users $\lambda_{mm}^{k+1},\ \forall m \in \mathcal{M}_n$, according to (17), after receiving the updated primal variables from clusters $C_j \in \mathcal{I}_{nm},\ \forall m \in \mathcal{M}_n$. Return to Step 1.

The dual updates are distributed by structure. The Lagrange multiplier λ_{nm}, corresponding to the SINR constraint of user U_{nm}, must be updated, at iteration k, according to $\lambda_{nm}^{k+1} = \lambda_{nm}^k + \tau\rho \sum_{j\in\mathcal{N}} \zeta_{jnm}^{k+1}$. This summation needs to include "influences" only from those clusters that exert non-negligible "influence" on U_{nm}, i.e., the set $\mathcal{I}_{nm} = \{C_j : U_{nm} \in \mathcal{B}_j, \forall j \in \mathcal{N}\}$.

We conclude this section with a remark on the stepsize parameter τ. According to the convergence analysis in [29], the primal stepsize τ in (16) must be determined as $\tau < \frac{2}{\max_{\{nm\}} |\mathcal{I}_{nm}|}$, where $|\cdot|$ denotes the cardinality of a set. Essentially, τ is affected by the number of clusters coupled in the "most populated" constraint in the system, i.e., the constraint corresponding to the user that suffers interference from the largest number of clusters. However here, the sets $\mathcal{I}_{nm}$ are not known a priori since they are determined by the optimal beamforming decisions. In this case, we need to determine conservative estimates for all the sets $\mathcal{I}_{nm}$, e.g., by letting each cluster's relays send maximum power pilot signals before the execution of the distributed algorithm. An analogous line of reasoning can be used to determine the communication neighborhood $\mathcal{C}_n$ of each cluster C_n, which in turn depends on appropriately defining the sets $\mathcal{B}_n$. Last, we note that, based on the analysis in [29], the penalty coefficient ρ must remain constant throughout the iterative execution. The Direct method is summarized in Alg. 1.

B. Indirect Method

The indirect method is also a 3-step primal-dual iterative scheme. Every cluster solves a local convex optimization problem, cf. (22), and, in the next two steps, it updates and transmits its primal, cf. (23), and dual variables, cf. (24), respectively. The main difference with the direct method lies in the way that we formulate the coupling constraints, cf. (18), which, in turn, leads to a different message exchange scheme. In particular, the indirect method allows us to manually define the message exchange network, cf. (19), without any dependencies on the inter-cluster interference patterns (see the pertinent discussion for the direct method in the end of Section III-A).

As with the direct method, the proposed indirect method also relies on defining appropriate auxiliary variables to introduce affine coupling constraints between the clusters. Consider, again, auxiliary variables ζ_{njm} exactly as described in (9) and (10) and $\boldsymbol{\zeta}_n = [\zeta_{n11}, \dots, \zeta_{nNM}]^T \in \mathbb{R}^{NM}$ and now also define $\boldsymbol{\zeta} = [\boldsymbol{\zeta}_1^T, \dots, \boldsymbol{\zeta}_N^T]^T \in \mathbb{R}^{N^2M}$ as the vector stacking all "influences" in the system. Furthermore, define local variables $\forall n, j \in \mathcal{N}$

$$\boldsymbol{\zeta}^{(n)} = [(\boldsymbol{\zeta}_1^{(n)})^T, \dots, (\boldsymbol{\zeta}_N^{(n)})^T]^T \in \mathbb{R}^{N^2L},$$

where $\boldsymbol{\zeta}_j^{(n)} = [\zeta_{j11}^{(n)}, \dots, \zeta_{jNK}^{(n)}]^T \in \mathbb{R}^{NL}$, so that each $\boldsymbol{\zeta}^{(n)}$ acts as an individual estimate of the global vector $\boldsymbol{\zeta}$ for every C_n and $\boldsymbol{\zeta}_j^{(n)}$ expresses the estimate that C_n has for the "influences" exerted by C_j on the system. The key idea behind this approach is to allow each decision maker C_n to maintain and update its own estimate of the global state of the system. Correspondingly, we need to enforce "consensus" among all these local variables, by imposing coupling, affine constraints of the form

$$\boldsymbol{\zeta}^{(1)} = \boldsymbol{\zeta}^{(2)} = \cdots = \boldsymbol{\zeta}^{(N)}. \tag{18}$$

There are many ways to express these equality constraints, depending on the message exchange capabilities between different clusters. In fact, let $G = (V, E)$ denote a directed graph defined on the set of clusters so that $V = \mathcal{N}$ is the set of vertices and $E \subseteq V \times V$ is the set of edges so that $E = \{(i,j) :\ j \in \mathcal{D}_i,\ i,j \in \mathcal{N}\}$. Here, $\mathcal{D}_i$ denotes the set of the 1-hop out-neighbors of node i in the graph G. Then, we can express (18) equivalently as

$$\begin{aligned} \boldsymbol{\zeta}^{(1)} &= \boldsymbol{\zeta}^{(j)}, \quad \forall j \in \mathcal{D}_1 \qquad (19)\\ &\vdots \\ \boldsymbol{\zeta}^{(N)} &= \boldsymbol{\zeta}^{(j)}, \quad \forall j \in \mathcal{D}_N \end{aligned}$$

if and only if the graph $G = (V, E)$ is weakly connected, i.e., if there exists an undirected path between any two nodes in the graph.

Using the coupling constraints (19), problem (8) can be transformed into

$$\min_{\{\mathbf{X}_n, \forall n\in\mathcal{N}\}} \sum_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n \mathbf{R}_T^n) \tag{20}$$

$$\begin{aligned} \text{s.t. } & \boldsymbol{\zeta}^{(n)} = \boldsymbol{\zeta}^{(i)}, \quad \forall n \in \mathcal{N},\ i \in \mathcal{D}_n \\ & \sum_{j\in\mathcal{N}} \boldsymbol{\zeta}_{jnm}^{(n)} \geq 0,\ \forall\, j, n \in \mathcal{N},\ m \in \mathcal{M}_n \\ & \zeta_{nnm}^{(n)} = \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{nnm}) - 1,\ \forall\, n \in \mathcal{N}, m \in \mathcal{M}_n \\ & \zeta_{njm}^{(n)} = \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{njm}),\ \forall\, n \in \mathcal{N}, j \in \mathcal{N}\backslash\{n\},\ m \in \mathcal{M}_j \\ & \mathbf{X}_n \in \mathbb{S}_+^L,\ \forall\, n \in \mathcal{N}. \end{aligned}$$

Again, all the constraints in (20) are local to each C_n and the only coupling constraints are the consistency constraints $\boldsymbol{\zeta}^{(n)} = \boldsymbol{\zeta}^{(i)},\quad \forall n \in \mathcal{N},\ i \in \mathcal{D}_n$, which are affine and thus amenable to decomposition by the ADAL algorithm.

Similar to the analysis in Section III-A, the augmented Lagrangian associated with (20) is

$$\Lambda(\{\mathbf{X}_n, \boldsymbol{\zeta}^{(n)}\}_{n=1}^N, \boldsymbol{\lambda}) = \underbrace{\sum_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n\mathbf{R}_T^n) + \sum_{n\in\mathcal{N}}\sum_{l\in\mathcal{D}_n} (\boldsymbol{\lambda}^{(nl)})^T(\boldsymbol{\zeta}^{(n)} - \boldsymbol{\zeta}^{(l)})}_{\text{Ordinary Lagrangian}} + \underbrace{\sum_{n\in\mathcal{N}}\sum_{l\in\mathcal{D}_n} \frac{\rho}{2}\|\boldsymbol{\zeta}^{(n)} - \boldsymbol{\zeta}^{(l)}\|_2^2}_{\text{Penalty term}},$$

where $\boldsymbol{\lambda}^{(nl)} \in \mathbb{R}^{N^2M}$ is the vector of Lagrange multipliers corresponding to the constraint $\boldsymbol{\zeta}^{(n)} = \boldsymbol{\zeta}^{(l)}$, defined for every $n \in \mathcal{N}$, $l \in \mathcal{D}_n$. As in Section III-A, we define local augmented Lagrangians $\forall n \in \mathcal{N}$

$$\begin{aligned}\Lambda_n(\mathbf{X}_n, \boldsymbol{\zeta}^{(n)}, \tilde{\boldsymbol{\zeta}}^{(j)}, \boldsymbol{\lambda}^{(nl)}) \; &= \; \mathrm{Tr}(\mathbf{X}_n\mathbf{R}_T^n) \\ &+ \sum_{l\in\mathcal{D}_n} (\boldsymbol{\lambda}^{(nl)})^T\boldsymbol{\zeta}^{(n)} + \sum_{\{m:n\in\mathcal{D}_m\}} (\boldsymbol{\lambda}^{(mn)})^T(-\boldsymbol{\zeta}^{(n)}) \\ &+ \sum_{l\in\mathcal{D}_n} \frac{\rho}{2}\|\boldsymbol{\zeta}^{(n)} - \tilde{\boldsymbol{\zeta}}^{(l)}\|_2^2 \; + \sum_{\{m:n\in\mathcal{D}_m\}} \frac{\rho}{2}\|\tilde{\boldsymbol{\zeta}}^{(m)} - \boldsymbol{\zeta}^{(n)}\|_2^2,\end{aligned} \tag{21}$$

where, again, the terms $\tilde{\boldsymbol{\zeta}}^{(l)}$ represent the local variables of each neighbor $C_l, l \in \mathcal{D}_n$ of C_n, that are communicated to C_n and considered constant with respect to the minimization of (21) at each respective iteration. Note that the term $\sum_{l\in\mathcal{D}_n}(\boldsymbol{\lambda}^{(nl)})^T\boldsymbol{\zeta}^{(n)} + \sum_{\{m:n\in\mathcal{D}_m\}}(\boldsymbol{\lambda}^{(mn)})^T(-\boldsymbol{\zeta}^{(n)})$ emerges from the consideration of the set of constraints (19).

Then, with every iteration k of the algorithm, every cluster C_n finds the minimizers $\hat{\boldsymbol{\zeta}}^{(n),k}$ of the local problem as

$$\begin{aligned}\hat{\boldsymbol{\zeta}}^{(n),k} = \quad & \mathrm{argmin}_{\{\mathbf{X}_n, \boldsymbol{\zeta}_n\}} \; \Lambda_n(\mathbf{X}_n, \boldsymbol{\zeta}^{(n)}, \tilde{\boldsymbol{\zeta}}^{(j),k}, \boldsymbol{\lambda}^k) \\ \text{s.t} \quad & \mathbf{X}_n \in \mathbb{S}_+^L, \; \boldsymbol{\zeta}^{(n)} \in Z_n,\end{aligned} \tag{22}$$

where, again, we define

$$Z_n = \Big\{\boldsymbol{\zeta}^{(n)} \in \mathbb{R}^{N^2M} | \zeta_{nnm}^{(n)} = \mathrm{Tr}(\mathbf{X}_n\mathbf{Q}^{nnm}) - 1, \forall m \in \mathcal{M}_n, \; \zeta_{njm}^{(n)} = \mathrm{Tr}(\mathbf{X}_n\mathbf{Q}^{njm}), \; \forall j \in \mathcal{N}\backslash\{n\}, \; m \in \mathcal{M}_j\Big\},$$

as the set of all points that satisfy the local constraints at each cluster. Subsequently, each cluster C_n updates its estimates $\tilde{\boldsymbol{\zeta}}^{(n),k}$ according to

$$\tilde{\boldsymbol{\zeta}}^{(n),k+1} = \tilde{\boldsymbol{\zeta}}^{(n),k} + \tau(\hat{\boldsymbol{\zeta}}^{(n),k} - \tilde{\boldsymbol{\zeta}}^{(n),k}), \tag{23}$$

and transmits the results to its in- and out-neighbors in the graph G. Finally, each cluster C_n updates its dual variables $\boldsymbol{\lambda}^{(nl),k+1}$, $\forall l \in \mathcal{D}_n$ according to

$$\boldsymbol{\lambda}^{(nl),k+1} = \boldsymbol{\lambda}^{(nl),k} + \tau\rho(\tilde{\boldsymbol{\zeta}}^{(n),k+1} - \tilde{\boldsymbol{\zeta}}^{(l),k+1}). \tag{24}$$

and transmits the updated values to its in- and out-neighbors in the graph G.

The intuition behind the indirect method is that each C_n tries to control its own variables, i.e., the entries $\zeta_{njm}^{(n)}$, $\forall j \in \mathcal{N}$, $m \in \mathcal{M}_j$, based on its current impression of the state of the system as expressed by the rest of the entries in $\boldsymbol{\zeta}^{(n)}$. As the iterations progress, the updated decisions of each cluster diffuse into the system and all clusters are forced to reach a consensus [38], in parallel with the optimization of the local utilities.

Algorithm 2 Indirect Method

Set $k = 0$ and define the consensus graph G, cf. (19). Every cluster C_n initializes and transmits its primal $\tilde{\boldsymbol{\zeta}}^{(n),0}$ and dual $\boldsymbol{\lambda}^{(nl),0}$ variables.

1. Every cluster C_n minimizes its local AL according to (22), after receiving the primal $\tilde{\boldsymbol{\zeta}}^{(l),k}$ and dual $\boldsymbol{\lambda}^{(l),k}$ variables from its in- and out-neighbors in the graph G.
2. Every cluster C_n updates and transmits its primal variables $\tilde{\boldsymbol{\zeta}}^{(n),k+1}$ according to (23).
3. Every cluster C_n updates and transmits its dual variables $\boldsymbol{\lambda}^{(nl),k+1}$, $\forall l \in \mathcal{D}_n$, according to (24), after receiving the updated primal variables $\tilde{\boldsymbol{\zeta}}^{(l),k+1}$ from its out-neighbors $l \in \mathcal{D}_n$. Return to Step 1.

Note that each $\tilde{\boldsymbol{\zeta}}^{(n)}$ does not necessarily need to include global information of the system. Instead, if we are able to estimate which users each cluster C_n will exert a negligible interference on, then we can neglect the corresponding entries of $\tilde{\boldsymbol{\zeta}}^{(n)}$ (and subsequently of $\boldsymbol{\lambda}^{(nl)}$), thus reducing the size of the problem significantly. Moreover, note that according to the convergence analysis of ADAL in [29], the choice of stepsize for the indirect method should be $\tau \leq 1/2$, because, for all possible communication graphs G, the coupling constraints will always involve only two decision makers (clusters). This is a direct consequence of the consensus constraints (19). The Indirect method is summarized in Alg. 2.

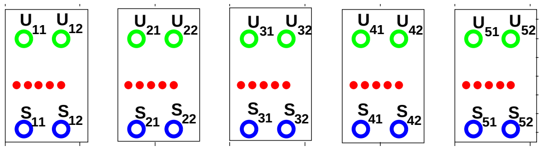
IV. Numerical Analysis

In this section, we illustrate the effectiveness of the proposed direct and indirect implementations of the ADAL algorithm for cooperative relay beamforming problems. We conducted simulations to examine the behavior of the proposed algorithms for different spatial configurations of the wireless networks and for various problem sizes. Comparative results with an existing distributed AL method, the Alternating Direction Method of Multipliers (ADMM) [32], are also presented. The ADMM is known to exhibit fast convergence speeds in general, for small though accuracies [32].

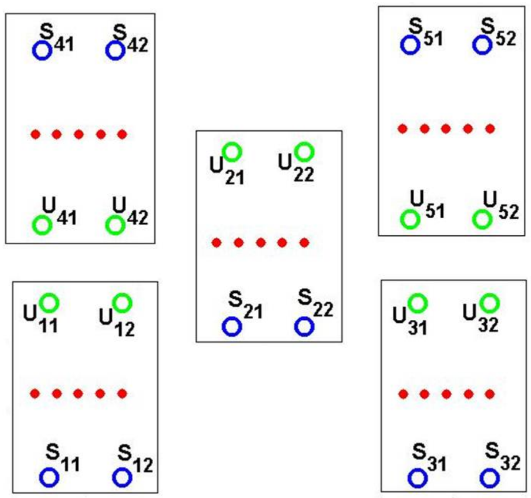
In all numerical experiments, we followed a channel model encompassing large scale fading effects due to path loss and small scale fading, i.e., we defined the channel between the source S_{nm} and relay R_{nl} as

$$\mathbf{f}_{nm,nl} = \alpha_{nm,nl} \; c_{nm,nl} \; e^{j(2\pi/\lambda)d_{nm,nl}} \;, \tag{25}$$

where $\alpha_{nm,nl}$ captures multipath fading, λ denotes the wavelength of carrier waves and $d_{nm,nl}$ denotes the Euclidean distance between the source S_{nm} and relay R_{nl}, and $c_{nm,nl} = d_{nm,nl}^{-\mu/2}$, where $\mu = 3.4$ is the path loss exponent and represents the power fall-off rate. Note that for simplicity, we did not include large-scale shadowing effects in (25), however,



(a)



(b)

Fig. 3. Two different spatial configurations of the multi-cluster network beamforming problem with: a) 5 clusters, and b) 5 clusters but larger interference levels compared to case (a), due to denser spatial positioning of the users. The blue and green circles correspond to sources and users, respectively, while the red dots depict the relays.

the extension is straightforward. Also, we assumed Rayleigh fading such that the gains $\alpha_{nm,nl}$ are i.i.d circularly symmetric complex Gaussian random variables with zero mean and unit variance, i.e., $\alpha_{nm,nl} \sim \mathcal{CN}(0,1)$. Correspondingly, for the purpose of simulations we constructed the channel state matrices by sampling realizations of Rayleigh random variables. The signal wavelength was assumed to be $\lambda = c/f = (3\cdot 10^8)/(2.4\cdot 10^9) = 0.125$m which is a reasonable choice for wireless transmissions utilizing ultra high frequency carrier waves (2.4GHz).

In all the cases presented below, we have set the initial values of the primal variables to 0, and randomly sampled the dual variables from a uniform distribution in $[0,1]$. Note that, different initialization values did not appear to affect the convergence speed significantly. Moreover, the penalty parameter ρ is in general user defined in augmented Lagrangian methods. In our simulations, we have found that fastest convergence is obtained for values $\rho \in [1,10]$, while at the same time preventing ill-conditioning.

In Fig. 4 we compare the two proposed methods, Direct and Indirect, for the 2 different setups of Fig. 3. Fig. 3(a) depicts a case with 5 clusters positioned in parallel, while Fig. 3(b) presents a case with 5 clusters but denser spatial positioning of the users. In both scenarios, we consider clusters containing 2 source-destination pairs and 5 relays, i.e., $|\mathcal{M}_n| = 2$ and $|\mathcal{L}_n| = 5$, $\forall\ n \in \mathcal{N}$, respectively. The same SINR requirement is set for all users at $\gamma = 10dB$. For the direct method, we assume that there exists at least one user that suffers non-negligible interference from all clusters, such that $\max_{\{nm\}} |\mathcal{I}_{nm}| = 5$, and hence we set $\tau = \frac{2}{5} = 0.4$; recall the pertinent discussion in section III-A. For the application of the indirect method, we model two cases: i) one which the available communication network between clusters is a simple line formation, so that we impose the coupling constraints

$$\boldsymbol{\zeta}^{(i)} = \boldsymbol{\zeta}^{(i+1)}, \qquad i = 1,2,\ldots,N-1,$$

and ii) a denser network with all-to-all communication between clusters, so that we impose the coupling constraints

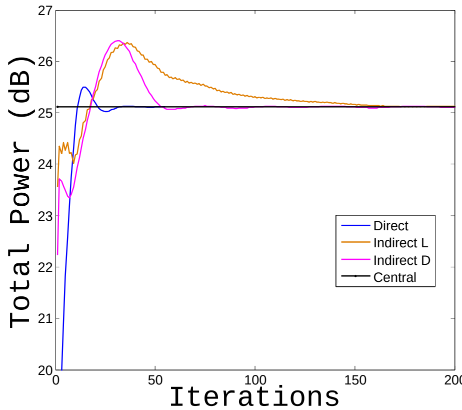
$$\boldsymbol{\zeta}^{(i)} = \boldsymbol{\zeta}^{(j)}, \qquad \forall\ \ i,j \in \mathcal{N}.$$

The idea behind considering two different indirect cases is to examine the effect of the communication network on the underlying consensus operations on the decision variables of the clusters. Note that each coupling constraint in the indirect method involves two decision makers, such that $\tau = \frac{1}{2}$ always; see also the pertinent discussion in section III-B. The simulation results in Fig. 4 show that in all cases the distributed ADAL algorithm leads to very fast convergence. Here, we note that the entries of the beamforming matrices obtained by ADAL converge to the respective values of the centralized solution. We also observe that the Indirect method is slower than the Direct one. Moreover, it is true that the Indirect method converges faster for denser communication networks, which is in accordance with the literature on consensus algorithms [38].

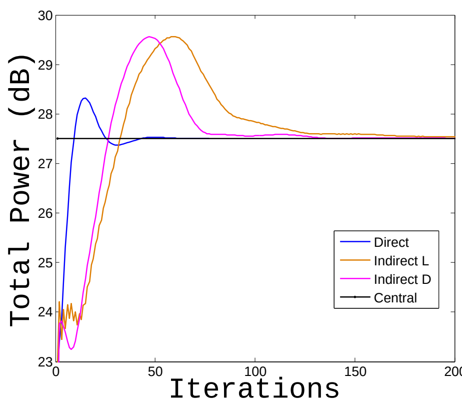
Fig. 4 also demonstrates how different system setups affect the speed of convergence. We observe that problems with a spatial configuration that induces higher interference levels, such as the setup of Fig. 3(b) compared to the setup of Fig. 3(a), tend to converge slightly slower. This is to be expected, since interference dominated scenarios will have SINR constraints that couple a relatively larger number of clusters, compared to cases with less interference for each user. This increased coupling naturally introduces the need for more coordination between the coupled decision makers (clusters), which leads to the slight increase in the number of iterations needed until convergence. To avoid confusion, we note that here we refer to the coupling between the beamforming decisions of all clusters due to the SINR constraints, and not the coupling from the consensus constraints (19) used in the indirect method.

Next, we compare our proposed distributed algorithm with the ADMM [32], which also utilizes augmented Lagrangians. Fig. 5 presents the results corresponding to application of the Direct method on both setups of Fig. 3.

Finally, in Fig. 6 we compare the two algorithms on a larger size problem of 15 clusters, with 3 blocks of the setup depicted in Fig. 3(b) positioned in parallel. In this case, we assume that each user suffers non-negligible interference from at most 10 other clusters, i.e, we take a safe estimate $\max_{\{nm\}} |\mathcal{I}_{nm}| = 10$, and hence we set $\tau = \frac{2}{10} = 0.2$. Fig. 6(b) contains the convergence results for the constraint violations $\sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n = \mathbf{0}$, i.e., how much the sum differs from zero. In all the scenarios considered, we can observe that the ADAL algorithm converges significantly faster than the respective ADMM. Note that, in all cases, we have used the



(a)



(b)

Fig. 4. Comparison of the Direct and Indirect methods for: a) the case depicted in Fig. 3(a), and b) the case depicted in Fig. 3(b). For the Indirect method we employed two different considerations of the constraint set (18), one for a line formation communication network between clusters (*Indirect L*) and one for an all-to-all communication network (*Indirect D*) as explained in text.

same initialization values for ADAL and ADMM. Moreover, after extensive sensitivity analysis in our simulations, we found that ADMM requires relatively larger values of $\rho \in [4, 40]$, compared to ADAL.

A. Discussion

As was shown above, the direct method converges faster than the indirect approach in general. Nevertheless, depending on the problem setup, it might be necessary that that indirect method is applied. For example, consider the two different setups of Fig. 3 and suppose the relays of each cluster are responsible to perform the necessary computations for the execution of ADAL. For the setup of Fig. 3(a), we can observe that if one cluster exerts interference to the users of a neighboring cluster, then most likely the corresponding relays are in range to exchange the necessary messages (due to the parallel spatial positioning of the clusters). However, the same does not hold true for setups where the relays of certain clusters may not be in communication range to exchange messages directly, even though they exert interference on each other's users. For instance, that would be the case between

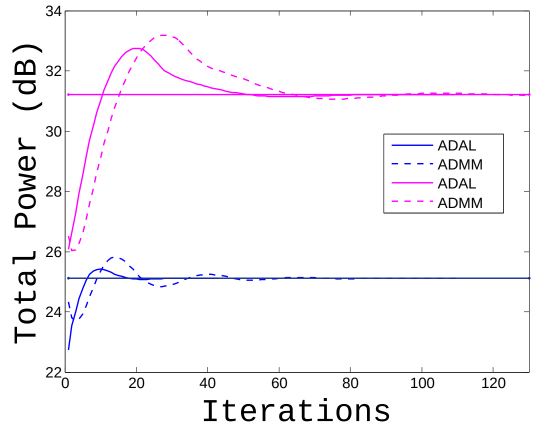
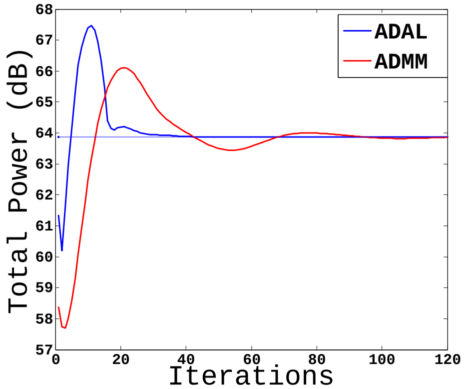


Fig. 5. Comparison of the two different distributed algorithms, ADAL and ADMM. The blue lines correspond to a problem with the setup of Fig. 3(a), while the pink lines correspond to a problem with the setup of Fig. 3(b). The results correspond to application of the Direct method.

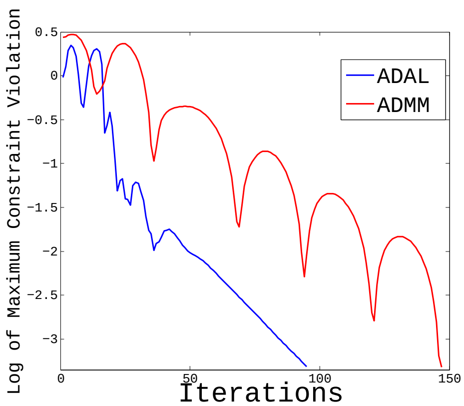
the cluster pairs $C_1 - C_4$ and $C_3 - C_5$ in Fig. 3(b). In such scenarios, we can employ the indirect method by defining the consensus constraint set (19) appropriately, such that there exists a feasible path of communication links between all clusters, e.g. for the setup of Fig. 3(b) this could be $\boldsymbol{\zeta}^{(1)} = \boldsymbol{\zeta}^{(2)}$, $\boldsymbol{\zeta}^{(2)} = \boldsymbol{\zeta}^{(3)}$, $\boldsymbol{\zeta}^{(4)} = \boldsymbol{\zeta}^{(2)}$, $\boldsymbol{\zeta}^{(5)} = \boldsymbol{\zeta}^{(2)}$. Alternatively, we can apply the direct method by allowing cluster C_2 to act as a "message relay" and convey the necessary message exchanges between the cluster pairs $C_1 - C_4$ and $C_3 - C_5$ at each iteration of ADAL. On the other hand, there might exist cases where there is no feasible communication path between all clusters, i.e., it is not possible to define a weakly connected graph E for the consensus constraints (19). In such scenarios, successful application of a distributed algorithm would require the users to act as "message relays" and convey the necessary message exchanges between the clusters.

V. Conclusions

We have considered the problem of cooperative beamforming in relay networks, for scenarios in which multiple clusters of source-destination node pairs, along with their dedicated relays, coexist in space. Since the original formulation leads to a non-convex problem, we have formulated an approximation of the problem in convex form and proposed a new, distributed optimization algorithm that allows for autonomous computation of the optimal beamforming decisions by each cluster, while taking into account intra- and inter-cluster interference effects. The advantage of the proposed approach is that it is a first order method utilizing augmented Lagrangians, thus it combines low computational complexity with the robustness and convergence speed properties of regularization. We have proposed two different ways of implementation and compared their relative performance. We have compared our method to a popular state-of-the-art distributed algorithm and showed that we obtain significant performance gains. To the best of the authors' knowledege, this is the first distributed solution for relay beamforming problems.



(a)



(b)

Fig. 6. Comparative convergence results for the ADAL and ADMM algorithms on a scenario with 15 clusters and $\gamma = 10$dB: a) Total transmitted power at the relays and b) Constraint feasibility evolution for the coupling constraints $\sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n = \mathbf{0}$. The results correspond to application of the Direct method.

References

[1] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity. part i. system description," *IEEE Trans. Comm.*, vol. 51, no. 11, pp. 1927–1938, Nov. 2003.

[2] A. B. Gershman, N. D. Sidiropoulos, S. Shahbazpanahi, and M. Bengtsson, "Convex optimization-based beamforming: From receive to transmit and network designs," *IEEE Signal Processing Magazine*, vol. 27, no. 3, p. 6275, May 2010.

[3] L. Dong, A. Petropulu, and H. Poor, "A cross-layer approach to collaborative beamforming for wireless ad hoc networks," *IEEE Trans. Signal Proc*, vol. 56, no. 7, pp. 2981–2993, July 2008.

[4] J. Li, A. Petropulu, and H. Poor, "Cooperative transmission for relay networks based on second-order statistics of channel state information," *IEEE Trans. Signal Proc*, vol. 59, no. 3, pp. 1280–1291, March 2011.

[5] L. Dong, A. Petropulu, and H. Poor, "Weighted cross-layer cooperative beamforming for wireless networks," *IEEE Trans. Signal Proc*, vol. 57, no. 8, pp. 3240–3252, Aug. 2009.

[6] Y. Liu and A. Petropulu, "Cooperative beamforming in multi-source multi-destination relay systems with SINR constraints," in *Proc. 2010 IEEE Conference on Acoustics Speech and Signal Processing (ICASSP)*, Dallas, TX, March 2010, pp. 2870–2873.

[7] Y. Liu and A. P. Petropulu, "On the sumrate of amplify-and-forward relay networks with multiple source-destination pairs," *IEEE Transactions on Wireless Communications*, vol. 10, no. 11, pp. 3732–3742, 2011.

[8] Y. Liu and A. Petropulu, "QoS guarantees in AF relay networks with multiple source-destination pairs in the presence of imperfect CSI," *Wireless Communications, IEEE Transactions on*, vol. 12, no. 9, pp. 4225–4235, September 2013.

[9] X. Li, Y. D. Zhang, and M. G. Amin, "Joint optimization of source power allocation and relay beamforming in multiuser cooperative wireless networks," *Mob. Netw. Appl.*, vol. 16, no. 5, pp. 562–575, Oct. 2011.

[10] H. Chen, A. Gershman, and S. Shahbazpanahi, "Distributed peer-to-peer beamforming for multiuser relay networks," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, april 2009, pp. 2265 –2268.

[11] M. Fadel, A. El-Keyi, and A. Sultan, "QoS-constrained multiuser peer-to-peer amplify-and-forward relay beamforming," *Signal Processing, IEEE Transactions on*, vol. 60, no. 3, pp. 1397–1408, March 2012.

[12] H. Q. Ngo and E. Larsson, "Large-scale multipair two-way relay networks with distributed af beamforming," *Communications Letters, IEEE*, vol. 17, no. 12, pp. 1–4, December 2013.

[13] N. Bornhorst, M. Pesavento, and A. Gershman, "Distributed beamforming for multi-group multicasting relay networks," *Signal Processing, IEEE Transactions on*, vol. 60, no. 1, pp. 221–232, Jan 2012.

[14] S. Fazeli-Dehkordy, S. Shahbazpanahi, and S. Gazor, "Multiple peer-to-peer communications using a network of relays," *Trans. Sig. Proc.*, vol. 57, no. 8, pp. 3053–3062, Aug. 2009.

[15] Y. Cheng and M. Pesavento, "Joint optimization of source power allocation and distributed relay beamforming in multiuser peer-to-peer relay networks," *Signal Processing, IEEE Transactions on*, vol. 60, no. 6, pp. 2962–2973, June 2012.

[16] C. Wang, H. Chen, Q. Yin, A. Feng, and A. Molisch, "Multi-user two-way relay networks with distributed beamforming," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3460–3471, October 2011.

[17] H. Kha, H. Tuan, and H. Nguyen, "Joint optimization of source power allocation and cooperative beamforming for SC-FDMA multi-user multi-relay networks," *Communications, IEEE Transactions on*, vol. 61, no. 6, pp. 2248–2259, June 2013.

[18] J. Choi, "MMSE-based distributed beamforming in cooperative relay networks," *Communications, IEEE Transactions on*, vol. 59, no. 5, pp. 1346 –1356, may 2011.

[19] Y. Jing and H. Jafarkhani, "Network beamforming using relays with perfect channel information," *IEEE Transactions on Information Theory*, vol. 55, no. 6, pp. 2499–2517, Jun. 2009.

[20] V. Havary-Nassab, S. Shahbazpanahi, A. Grami, and Z.-Q. Luo, "Distributed beamforming for relay networks based on second-order statistics of the channel state information," *Trans. Sig. Proc.*, vol. 56, no. 9, pp. 4306–4316, Sep. 2008.

[21] Z. Ding, W. H. Chin, and K. Leung, "Distributed beamforming and power allocation for cooperative networks," *Wireless Communications, IEEE Transactions on*, vol. 7, no. 5, pp. 1817 –1822, may 2008.

[22] P. Gupta and P. Kumar, "The capacity of wireless networks," *IEEE Trans. Inform. Theory*, vol. 46, no. 2, pp. 388–404, 2000.

[23] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, September 2004.

[24] M. Zavlanos, A. Ribeiro, and G. Pappas, "Network integrity in mobile robotic networks," *Automatic Control, IEEE Transactions on*, vol. 58, no. 1, pp. 3–18, Jan 2013.

[25] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Review*, vol. 38, no. 1, pp. 49–95, 1996.

[26] Y. Huang and D. Palomar, "Rank-constrained separable semidefinite programming with applications to optimal beamforming," *IEEE Trans. on Signal Processing*, vol. 58, no. 2, p. 664678, Feb. 2010.

[27] D. Hammarwall, M. Bengtsson, and B. Ottersten, "On downlink beamforming with indefinite shaping constraints," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3566–3580, Sep. 2006.

[28] M. Bengtsson and B. Ottersten, "Optimal downlink beamforming using semidefinite optimization," in *Proc. of 37th Annual Allerton Conference on Communication, Control, and Computing*, 1999, pp. 987–996.

[29] N. Chatzipanagiotis, D. Dentcheva, and M. Zavlanos, "An augmented Lagrangian method for distributed optimization," *Mathematical Programming*, 2014.

[30] A. Ruszczynski, *Nonlinear Optimization*. Princeton, NJ, USA: Princeton University Press, 2006.

[31] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[32] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[33] J. Mulvey and A. Ruszczynski, "A diagonal quadratic approximation method for large scale linear programs," *Operations Research Letters*, vol. 12, pp. 205–215, 1992.

[34] A. Tolli, H. Pennanen, and P. Komulainen, "Distributed coordinated multi-cell transmission based on dual decomposition," in *IEEE GLOBECOM 2009*, Nov. 2009, pp. 1–6.
[35] C. Shen, T.-H. Chang, K.-Y. Wang, Z. Qiu, and C.-Y. Chi, "Distributed robust multi-cell coordinated beamforming with imperfect CSI: An ADMM approach," *IEEE Trans. on Signal Processing*, vol. 60, no. 6, p. 29883003, Jun. 2012.
[36] R. Gallager, *Principles of Digital Communication*. Cambridge Univ. Press, Cambridge, UK, 2008.
[37] N. Sidiropoulos, T. Davidson, and Z. Luo, "Transmit beamforming for physical-layer multicasting," *IEEE Trans. Signal Proc*, vol. 54, no. 6, pp. 2239–2251, June 2006.
[38] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 169–182, February 2005.

Nikolaos Chatzipanagiotis received the Diploma in Mechanical Engineering, and the M.Sc. degree in Microsystems and Nanodevices from the National Technical University of Athens, Athens, Greece, in 2006 and 2008, respectively. Currently, he is working toward the Ph.D. degree in the Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC. From 2010 to 2012, he was a graduate student in the Department of Mechanical Engineering at the Stevens Institute of Technology, Hoboken, NJ. He was the recipient of the 2012-2013 graduate study scholarship from the Gerondelis Foundation, and the 2011-2012 I&E doctoral fellowship from the Stevens Institute of Technology.

His research interests include distributed and stochastic optimization algorithms with applications on networked control systems, wired and wireless communications, and multi-agent mobile robotic networks.

Yupeng Liu (M'09) received his B.S. degree in 2005 and M.S. degree in 2008 from University of Science and Technology of China and Ph.D. degree in 2012 from Rutgers, the State University of New Jersey, all in Electrical Engineering. He is currently with Alcatel-Lucent, New Providence, NJ, USA.

His research interests include multiple user wireless communications, cooperative communications and physical layer secrecy.

Athina Petropulu received her undergraduate degree from the National Technical University of Athens, Greece, and the M.Sc. and Ph.D. degrees from Northeastern University, Boston MA, all in Electrical and Computer Engineering. Between 1992-2010 she was faculty at the Department of Electrical and Computer Engineering at Drexel University. Since 2010, she is Professor and Chair of the Electrical and Computer Engineering Department at Rutgers, The State University of New Jersey. She has held visiting appointments at SUPELEC in France and at Princeton University. Dr. Petropulu's research interests span the area of statistical signal processing with applications in wireless communications, networking and radar systems. She is recipient of the 1995 Presidential Faculty Fellow Award in Electrical Engineering given by NSF and the White House. She is the co-author (with C.L. Nikias) of the textbook entitled, "Higher-Order Spectra Analysis: A Nonlinear Signal Processing Framework," (Englewood Cliffs, NJ: Prentice-Hall, Inc., 1993).

Dr. Petropulu is Fellow of IEEE. She has served as Editor-in-Chief of the IEEE Transactions on Signal Processing (2009-2011), IEEE Signal Processing Society Vice President-Conferences (2006-2008), and member-at-large of the IEEE Signal Processing Board of Governors. She was the General Chair of the 2005 International Conference on Acoustics Speech and Signal Processing (ICASSP-05), Philadelphia PA. She is co-recipient of the 2005 IEEE Signal Processing Magazine Best Paper Award, and the 2012 Meritorious Service Award from the IEEE Signal Processing Society.

Michael M. Zavlanos (S'05-M'09) received the Diploma in mechanical engineering from the National Technical University of Athens (NTUA), Athens, Greece, in 2002, and the M.S.E. and Ph.D. degrees in electrical and systems engineering from the University of Pennsylvania, Philadelphia, PA, in 2005 and 2008, respectively.

From 2008 to 2009 he was a Post-Doctoral Researcher in the Department of Electrical and Systems Engineering at the University of Pennsylvania, Philadelphia. He then joined the Stevens Institute of Technology, Hoboken, NJ, as an Assistant Professor of Mechanical Engineering, where he remained until 2012. Currently, he is an assistant professor of mechanical engineering and materials science at Duke University, Durham, NC. He also holds a secondary appointment in the department of electrical and computer engineering. His research interests include a wide range of topics in the emerging discipline of networked systems and science, with applications in robotic, sensor, communication, and biomolecular networks. He is particularly interested in hybrid solution techniques, on the interface between control theory, distributed optimization, estimation, and networking.

Dr. Zavlanos is a recipient of the 2014 Office of Naval Research Young Investigator Program (YIP) Award, the 2011 National Science Foundation Faculty Early Career Development (CAREER) Award, and a finalist for the Best Student Paper Award at CDC 2006.