A Distributed Algorithm for Cooperative Relay Beamforming

Nikolaos Chatzipanagiotis, Athina Petropulu, Michael M. Zavlanos

Abstract— **We consider the problem of cooperative beamforming in relay networks. Assuming knowledge of the second-order statistics of channel state information (CSI), the optimal beamforming weights are determined so that the total transmitted power at the relays is minimized, while meeting signal-to-interference-plus-noise-ratio (SINR) requirements at the destinations. Our formulation utilizes the Semidefinite Relaxation (SDR) technique to produce an efficient, tractable convex programming approximation of the problem in centralized form. We propose a distributed optimization algorithm to calculate the optimal beamforming weights for scenarios where multiple clusters of source-destination node pairs, along with their dedicated relays, coexist in space. Our method relies on dual decomposition techniques with regularization, that can significantly improve on the inherent disadvantages of simple dual subgradient methods. Numerical anaysis demonstrates that the proposed algorithm exhibits very fast convergence rates.**

I. Introduction

Cooperative relay beamforming [1]–[5] is a rapidly emerging area of interest in the field of multiantenna smart signaling strategies, due to its potential to provide energy efficiency and communication reliability in long distance transmissions, where signal fading is a limiting factor. Traditionally, multihop schemes have been utilized to address such scenarios. However real-time transmissions over multiple hops suffer from packet collisions and interference, thus introducing long delays, especially so for ad hoc networks [6].

A preferred solution to these issues is relay beamforming, wherein a set of relay nodes cooperate to form a "virtual antenna array" that retransmits signals from sources to destinations. The main idea behind this technique is to exploit constructive interference effects, by forming beampatterns that focus on the destinations' locations. This results in increased directional channel gain [7, 8] and enables long distance transmissions with lower power and fewer hops, minimizing thus interference [6, 9]. In this paper, we focus on the very popular Amplify-and-Forward (AF) relaying protocol, due to its low complexity and implementation cost [4]. In AF the relay antennas retransmit an amplified and phase-steered version of the received source signals, by multiplying them with appropriate, optimal weights (linear precoders). The optimality criterion to determine these weights typically involves minimizing the total transmitted power at the relays, subject to satisfying desirable SINR levels at all destinations.

This work is partially supported by NSF under grants CNS#1239339, CNS#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. 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.

Beamforming algorithms require access to channel state information (CSI), upon which the calculation of the optimal beamforming weights is based. Thus, centralized optimization methods, which involve a central processing unit that collects the global CSI information and then transmits the optimal weights to all beamformers, incur a large communication cost. They also entail significant delays and are vulnerable to failures, giving rise to the need for distributed techniques, where each beamformer must calculate its optimal weights based on local information only. In this paper, we consider scenarios where multiple clusters of source-destination pairs, along with their dedicated relays, coexist in space. We propose a novel distributed optimization algorithm, which allows for autonomous computation of the beamforming decisions by each cluster, while taking into account intra- and inter-cluster interference effects.

Our algorithm utilizes *augmented Lagrangians* (AL), a regularization technique that is obtained by adding a quadratic penalty term to the ordinary Lagrangian [10, 11]. AL methods converge very fast, especially compared to first order methods, however they lack the decomposability properties of the ordinary Lagrangian, utilized in the well known *dual decomposition* method [11]. Nevertheless, simple dual decomposition methods suffer from slow convergence rates and require strict convexity of the objective function [10]–[12]. The later strict convexity requirement prohibits application of such techniques to the problem under consideration, as the proposed objective function is linear. In this paper, we employ the *Accelerated Distributed Augmented Lagrangians* (ADAL) algorithm that we recently proposed in [13]. ADAL is a novel AL decomposition method that enjoys very fast convergence rates, as suggested by numerical simulations. Alternative AL decomposition techniques for general convex optimization problems can be found in [14]–[18]. In relevant beamforming literature, distributed methods for the multicell downlink beamforming problem have been proposed in [19], utilizing the dual decomposition method, and also in [20], where an AL method is used, namely the *Alternating Directions Method of Multipliers* (ADMM). To the best of the authors knowledge, distributed algorithms for the case of multi-cluster relay beamforming, as discussed here, have not been proposed.

The rest of the paper is organized as follows: In Section II, we discuss the multi-cluster network beamforming problem and formulate it as a convex optimization problem. In Section III, we propose a distributed algorithm to solve the aforementioned problem based on ADAL. Finally, in Section IV, we present numerical results to verify the validity of our approach.

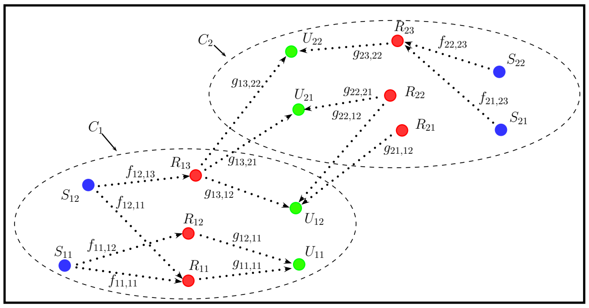


Fig. 1. A $N = 2$ multi-cluster relay beamforming scenario, with $M = 2$ source-destination pairs (green and blue dots respectively) and $L = 3$ dedicated relays (red dots) for each cluster.

II. Cooperative Relay Beamforming

Consider a scenario where a set $\mathcal{N} = \{1, \ldots, N\}$ of clusters coexist in space, where each cluster $C_n, \forall n \in \mathcal{N}$ is 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}, $\forall n \in \mathcal{N}, l \in \mathcal{L}_n$. A simple case is depicted in Fig. 1. 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. We consider cases where the direct communication links between source-destination pairs are of prohibitively low quality, i.e. the channel gains are negligible, such that utilization of the relays is justifiable. The received signal at every relay R_{nl} is given by

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

where $\mathbb{C}$ denotes the complex line, 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, $v_{nl} \in \mathbb{C}$ is the noise at relay R_{nl} modeled as i.i.d. circularly symmetric, complex Gaussian random variable with unit variance, i.e. $v_{nl} \sim \mathcal{CN}(0, 1)$ and $f_{jm,nl}$ denotes the channel gain between source S_{jm} and relay R_{nl}. 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 $\mathbf{s}_j = [s_{j1}, \ldots, s_{jM}] \in \mathbb{C}^M$, $\mathbf{x}_n = [x_{n1}, \ldots, x_{nL}]^T \in \mathbb{C}^L$, $\mathbf{v}_n = [v_{n1}, \ldots, v_{nL}]^T \in \mathbb{C}^L$. 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 C_j to all the relays of C_n, i.e.

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

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. During the second communication stage the relays of cluster C_n retransmit, in an AF fashion, a linear transformation of $\mathbf{x}_n$, 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. In this paper, we consider the case where every relay node carries a single antenna, which translates into the beamforming matrix being diagonal, i.e. $\mathbf{W}_n = \mathrm{diag}\{w_{n1}, \ldots, w_{nL}\} \in \mathbb{C}^{L \times L}$, where w_{nl} denotes the complex weight with which relay R_{nl} multiplies its received signal. These beamforming decisions of the relays of each cluster must also take into account interference effects at the intended users caused by the other clusters' operation. Then, the received signal vectors $\mathbf{y}_n \in \mathbb{C}^M$ at all users of each cluster C_n will be

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

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.

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

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}. More specifically, the received signal at user U_{nm} is comprised of

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

Subsequently, the SINR of user U_{nm} is given by

$$\begin{aligned} \mathrm{SINR}_{nm} = & \, \mathbb{E}\Big(P_0 |\mathbf{g}_{n,nm}^T \mathbf{W}_n \mathbf{f}_{nm,n} s_{nm}|^2 \Big) / \\ & \mathbb{E}\Big(P_0 \sum_{i \in \mathcal{M}_n}^{i \neq m} |\mathbf{g}_{n,nm}^T \mathbf{W}_n \mathbf{f}_{ni,n} s_{ni}|^2 + \sum_{j \in \mathcal{N}} |\mathbf{g}_{j,nm}^T \mathbf{W}_j \mathbf{v}_j|^2 \\ & + P_0 \sum\nolimits_{j \in \mathcal{N}}^{j \neq n} \sum\nolimits_{k \in \mathcal{M}_j} |\mathbf{g}_{n,nm}^T \mathbf{W}_n \mathbf{f}_{jk,n} \mathbf{s}_{jk}|^2 \\ & + P_0 \sum_{j \in \mathcal{N}}^{j \neq n} \sum_{i \in \mathcal{N}} \sum_{k \in \mathcal{M}_i} |\mathbf{g}_{j,nm}^T \mathbf{W}_j \mathbf{f}_{ik,j} \mathbf{s}_{ik}|^2 + |z_{nm}|^2 \Big) \end{aligned}$$

with $\mathbb{E}(\cdot)$ denoting expectation with respect to time. As already mentioned, a reasonable optimality criterion, according to which we calculate the optimal beamforming weights, involves minimizing the total transmitted power at the relays, subject to satisfying user-specific SINR lower bounds $\gamma_{nm} > 0$. Thus, the multi-cluster beamforming problem entails finding $\mathbf{W}_n$ that solve the optimization problem

$$\min_{\{\mathbf{W}_n\}n\in\mathcal{N}} \quad \sum\nolimits_{n\in\mathcal{N}} P_T^n(\mathbf{W}_n) \quad (1)$$
$$\text{s.t. } \mathrm{SINR}_{nm}(\mathbf{W}_n) \geq \gamma_{nm}, \ \ \forall n \in \mathcal{N}, \ m \in \mathcal{M}_n$$

where the average, total transmited power at the relays of cluster C_n is calculated as $P_T^n = \mathbb{E}\{\|\mathbf{t}_n\|_F^2\} = \sum_{j\in\mathcal{N}} \mathrm{Tr}\big(P_0\mathbf{W}_n\mathbb{E}\{\mathbf{F}_{jn}\mathbf{F}_{jn}^{\mathcal{H}}\}\mathbf{W}_n^{\mathcal{H}}\big) + \mathrm{Tr}\big(\mathbf{W_n}\mathbf{W_n^{\mathcal{H}}}\big)$, with $\|\cdot\|_F$ denoting the Frobenius norm. In this paper, we assume that every relay node carries a single antenna, which translates to every $\mathbf{W}_n$ being a diagonal matrix. This enables us to express the total transmit power of C_n 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 $\mathbf{R}_T^n = \mathbf{I}_L + P_0 \sum_{j\in\mathcal{N}} \sum_{m\in\mathcal{M}_j} \mathrm{diag}\Big\{\mathbb{E}\{|f_{jm,n1}|^2\}, \ldots, \mathbb{E}\{|f_{jm,nL}|^2\}\Big\}$, with $\mathbf{I}_L$ denoting the identity matrix of size L. Additionally, we define $\forall n \in \mathcal{N}, m \in \mathcal{M}_n$ the desired signal matrices as $\mathbf{R}_S^{nm} = \mathbb{E}\{(\mathbf{f}_{nm,n}^T \odot \mathbf{g}_{n,nm}^T)^{\mathcal{H}}(\mathbf{f}_{nm,n}^T \odot \mathbf{g}_{n,nm}^T)\}$, where $\odot$ denotes the Hadamard (entrywise) product. The intra-cluster interference matrices are $\mathbf{R}_I^{nm} = \sum_{i\in\mathcal{M}_n}^{i\neq m} \mathbb{E}\{(\mathbf{f}_{ni,n}^T \odot \mathbf{g}_{n,nm}^T)^{\mathcal{H}}(\mathbf{f}_{ni,n}^T \odot \mathbf{g}_{n,nm}^T)\} + \sum_{j\in\mathcal{N}}^{j\neq n}\sum_{k\in\mathcal{M}_j} \mathbb{E}\{(\mathbf{f}_{jk,n}^T \odot \mathbf{g}_{n,nm}^T)^{\mathcal{H}}(\mathbf{f}_{jk,n}^T \odot \mathbf{g}_{n,nm}^T)\}$. The inter-cluster interference matrices $\forall j \in \mathcal{N}\backslash\{n\}$ are $\mathbf{R}_{IC}^{j,nm} = \sum_{i\in\mathcal{N}}\sum_{k\in\mathcal{M}_i} \mathbb{E}\{(\mathbf{f}_{ik,j}^T \odot \mathbf{g}_{j,nm}^T)^{\mathcal{H}}(\mathbf{f}_{ik,j}^T \odot \mathbf{g}_{j,nm}^T)\}$ and, finally, the noise matrices are $\mathbf{R}_v^{j,nm} = \mathrm{diag}\big\{\mathbb{E}\{|g_{j1,nm}|^2\}, \ldots, \mathbb{E}\{|g_{jL,nm}|^2\}\big\}$. Using this notation, the SINR_{nm} is expressed as

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

Then, problem (1) can be equivalently written as

$$\begin{aligned} \min_{\{\mathbf{w}_n\},n\in\mathcal{N}} \quad & \sum\nolimits_{n\in\mathcal{N}} \mathbf{w}_n^{\mathcal{H}} R_T^n \mathbf{w}_n \\ \text{s.t.} \quad & \mathbf{w}_n^{\mathcal{H}}\mathbf{Q}^{nnm}\mathbf{w}_n + \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \mathbf{w}_j^{\mathcal{H}}\mathbf{Q}^{jnm}\mathbf{w}_j \geq 1, \\ & \forall n \in \mathcal{N}, \ m \in \mathcal{M}_n, \end{aligned} \quad (2)$$

where we have further defined the matrices $\mathbf{Q}^{nnm} = \frac{P_0}{\gamma_{nm}}\mathbf{R}_S^{nm} - P_0\mathbf{R}_I^{nm} - \mathbf{R}_v^{n,nm}$ and $\mathbf{Q}^{jnm} = -P_0\mathbf{R}_{IC}^{j,nm} - \mathbf{R}_v^{j,nm}$. The matrices $\mathbf{Q}^{ijk}$ are Hermitian, as the sums of Hermitian matrices, and will be, in general, indefinite. This means that the optimization problem (2) belongs in the class of nonconvex Quadratically Constrained Quadratic Programming (QCQP) problems, which are NP-hard to solve. Nevertheless, by defining the variable $\mathbf{X}_n \triangleq \mathbf{w}_n\mathbf{w}_n^{\mathcal{H}}$ [21] we can express (2) in the equivalent form:

$$\begin{aligned} \min_{\{\mathbf{X}_n\},n\in\mathcal{N}} \quad & \sum\nolimits_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n\mathbf{R}_T^n) \\ \text{s.t.} \quad & \mathrm{Tr}(\mathbf{X}_n\mathbf{Q}^{nnm}) + \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \mathrm{Tr}(\mathbf{X}_j\mathbf{Q}^{jnm}) \geq 1, \\ & \mathbf{X}_n \in \mathbb{S}_+^L, \quad \forall n \in \mathcal{N}, \ m \in \mathcal{M}_n, \\ & \mathrm{rank}(\mathbf{X}_n) = 1, \quad \forall n \in \mathcal{N} \end{aligned} \quad (3)$$

where $\mathbf{X}_n \in \mathbb{S}_+^L$ imposes the (convex) constraint that matrix $\mathbf{X}_n$ belongs to the cone of symmetric, positive semidefinite matrices of dimension L. Problem (3) is equivalent to (2) and still nonconvex because of the nonconvex rank constraint. Utilizing the so-called *Semidefinite Relaxation* (SDR) technique [21], we can drop the rank constraints (thus enlarging the feasible set) in (3) and solve:

$$\begin{aligned} \min_{\{\mathbf{X}_n\},n\in\mathcal{N}} \quad & \sum\nolimits_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n\mathbf{R}_T^n) \\ \text{s.t.} \quad & \mathrm{Tr}(\mathbf{X}_n\mathbf{Q}^{nnm}) + \sum\nolimits_{j\in\mathcal{N}}^{j\neq n} \mathrm{Tr}(\mathbf{X}_j\mathbf{Q}^{jnm}) \geq 1, \\ & \mathbf{X}_n \in \mathbb{S}_+^L, \quad \forall n \in \mathcal{N}, \ m \in \mathcal{M}_n, \end{aligned} \quad (4)$$

at the cost of obtaining, possibly, suboptimal solutions. The advantage of (4) is that it is a semidefinite programming problem and can be efficiently solved by interior point methods. Due to the relaxation, the optimizer $\mathbf{X}_n^*$ of (4) will not be rank one in general. If it is, then it will be the optimal solution to the original problem (3). If not, randomization techniques [22] can be employed to obtain a rank one matrix.

Remark 1 *Observe that, similar to [2]–[5], we assume knowledge of the second order statistics of CSI, which in a practical setting can be obtained based on past observations. Also, we define the SINR 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. [23] and in published works related to the problem considered here [2]–[5]. A scenario with perfect channel knowledge would express the constraints with respect to instantaneous SINRs, however this consideration lacks in terms of practical applicability.*

III. Distributed Optimization

Dual subgradient methods that exploit the separability of the ordinary Lagrangian are simple and attractive, however, they are not suitable for the problem under consideration, since they require strict convexity of the objective function and also suffer from very slow convergence rates. This motivates alternative methods that take advantage of regularization techniques, such as the *augmented Lagrangian Method* (ALM) [10, 11], which we briefly present in what follows.

As our distributed version of ALM is developed for affine equality constraints, we define auxilliary variables $\zeta_{njm}, \forall n, j \in \mathcal{N}, m \in \mathcal{M}_n$ that express the amount of 'influence' (meaning either the desired signal power or interference) exerted by all the relays of cluster C_n on user

Algorithm 1 Augmented Lagrangian Method

Require: Set iteration counter $k = 1$ and define initial Lagrange multipliers $\boldsymbol{\lambda}^1$.

1: For fixed Lagrange multipliers $\boldsymbol{\lambda}^k$, find primal variables $\mathbf{X}_n^k, \boldsymbol{\zeta}_n^k$ that solve the problem:

$$\{\mathbf{X}^k, \boldsymbol{\zeta}^k\} = \arg\min_{\mathbf{X},\boldsymbol{\zeta}} \Lambda(\mathbf{X}, \boldsymbol{\zeta}, \boldsymbol{\lambda}^k) \quad \text{s.t.} \quad \mathbf{X}_n \in \mathbb{S}_+^L,\ \boldsymbol{\zeta}_n \in Z_n, \quad \forall\, n \in \mathcal{N} \tag{8}$$

2: If the constraints $\sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n^k = \mathbf{0}$ are satisfied, then stop (optimal solution found). Otherwise, set :

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \rho \sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n^k \tag{9}$$

increase k by one and return to Step 1.

U_{jm}, i.e.

$$\begin{aligned} \zeta_{nnm} &= \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{nnm}) - 1, \quad \forall m \in \mathcal{M}_n \\ \zeta_{njm} &= \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{njm}), \qquad \forall\, j \in \mathcal{N}\backslash\{n\},\ m \in \mathcal{M}_j \end{aligned} \tag{5}$$

Furthermore, define the vector $\boldsymbol{\zeta}_n = [\zeta_{n11}, \ldots, \zeta_{nNM}]^T \in \mathbb{R}^{NM}$ stacking all the 'influences' of C_n. Now, note that the inequality constraints in (4) 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. Then, problem (4) can be equivalently written as

$$\begin{aligned} \min_{\{\mathbf{X}_n\}, n\in\mathcal{N}} \quad & \sum_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n \mathbf{R}_T^n) \\ \text{s.t.} \quad & \sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n = \mathbf{0} \\ & \zeta_{nnm} = \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{nnm}) - 1, \quad \forall n \in \mathcal{N}\ m \in \mathcal{M}_n \\ & \zeta_{njm} = \mathrm{Tr}(\mathbf{X}_n \mathbf{Q}^{njm}), \forall n \in \mathcal{N}, j \in \mathcal{N}\backslash\{n\}, m \in \mathcal{M}_j \\ & \mathbf{X}_n \in \mathbb{S}_+^L, \quad \forall n \in \mathcal{N} \end{aligned} \tag{6}$$

where $\mathbf{0}$ is the zero vector of dimension NM. The augmented Lagrangian associated with (6) is

$$\Lambda(\mathbf{X}, \boldsymbol{\zeta}, \boldsymbol{\lambda}) = \overbrace{\sum_{n\in\mathcal{N}} \mathrm{Tr}(\mathbf{X}_n \mathbf{R}_T^n) + \boldsymbol{\lambda}^T \sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n}^{\text{Ordinary Lagrangian}} + \overbrace{\frac{\rho}{2} \| \sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n \|_2^2}^{\text{Penalty term}} \tag{7}$$

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 (7), because the rest of the constraints are local at each cluster C_n and, hence, are not included in the AL. In what follows, for simplicity of notation, we collectively denote the set of points satisfying the local constraints of each cluster C_n as

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

Algorithm 2 Accelerated Distributed AL (ADAL)

Require: Set $k = 1$ and define initial Lagrange multipliers $\boldsymbol{\lambda}^1$ and primal variables $\tilde{\boldsymbol{\zeta}}_n^1$.

1: For fixed $\boldsymbol{\lambda}^k$, $\tilde{\boldsymbol{\zeta}}_n^k$ calculate for all clusters $C_n, n \in \mathcal{N}$ the $\hat{\boldsymbol{\zeta}}_n^k$ as the solution of

$$\begin{aligned} \arg\min_{\mathbf{X}_n, \boldsymbol{\zeta}_n} \quad & \Lambda_n(\mathbf{X}_n, \boldsymbol{\zeta}_n, \{\tilde{\boldsymbol{\zeta}}_j^k\}_{j\in\mathcal{C}_n}, \boldsymbol{\lambda}^k) \\ \text{s.t.} \quad & \mathbf{X}_n \in \mathbb{S}_+^L,\ \boldsymbol{\zeta}_n \in Z_n \end{aligned} \tag{10}$$

2: If $\sum_{n\in\mathcal{N}} \tilde{\boldsymbol{\zeta}}_n^k = \mathbf{0}$, then stop (optimal solution found). Otherwise, for $n \in \mathcal{N}$ set

$$\tilde{\boldsymbol{\zeta}}_n^{k+1} = \tilde{\boldsymbol{\zeta}}_n^k + \tau(\hat{\boldsymbol{\zeta}}_n^k - \tilde{\boldsymbol{\zeta}}_n^k) \tag{11}$$

and communicate $\tilde{\boldsymbol{\zeta}}_n^{k+1}$ to every $C_j \in \mathcal{C}_n$.

3: For every $n \in \mathcal{N}$ and every $m \in \mathcal{M}_n$ set

$$\lambda_{nm}^{k+1} = \lambda_{nm}^k + \tau\rho \sum_{j\in\mathcal{I}_{nm}} \tilde{\zeta}_{jnm}^{k+1} \tag{12}$$

communicate λ_{nm}^{k+1} to every $C_j \in \mathcal{I}_{nm}$, increase k by 1 and go to Step 1.

Alg. 1 summarizes the centralized version of the augmented Lagrangian method applied on our problem for reference. For a more detailed discussion see [10, 11].

A. Accelerated Distributed Augmented Lagrangians

The ALM is an excellent general purpose method, however (7) is not separable, due to the quadratic penalty term and the underlying inner products $\langle \boldsymbol{\zeta}_i, \boldsymbol{\zeta}_j \rangle$ that form, when this term is expanded. Thus, in order to obtain a distributed method, we use the *Accelerated Distributed Augmented Lagrangians* (ADAL) algorithm that we have recently proposed in [13]. Implementation of ADAL involves defining *local* AL for every cluster C_n

$$\Lambda_n(\mathbf{X}_n, \boldsymbol{\zeta}_n, \{\tilde{\boldsymbol{\zeta}}_j\}_{j\in\mathcal{N}}^{j\neq n}, \boldsymbol{\lambda}) = \mathrm{Tr}(\mathbf{X}_n \mathbf{R}_T^n) + \boldsymbol{\lambda}^T \boldsymbol{\zeta}_n + \frac{\rho}{2} \| \boldsymbol{\zeta}_n + \sum_{j\in\mathcal{N}}^{j\neq n} \tilde{\boldsymbol{\zeta}}_j \|_2^2$$

where we introduce $\tilde{\boldsymbol{\zeta}}_j$, 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. At each iteration k of ADAL, each cluster C_n minimizes its local augmented Lagrangian

$$\hat{\boldsymbol{\zeta}}_n^k = \mathrm{argmin}_{\{\mathbf{X}_n \in \mathbb{S}_+^L, \boldsymbol{\zeta}_n \in Z_n\}} \Lambda_n(\mathbf{X}_n, \boldsymbol{\zeta}_n, \{\tilde{\boldsymbol{\zeta}}_j^k\}_{j\in\mathcal{N}}^{j\neq n}, \boldsymbol{\lambda}^k) \tag{13}$$

A key observation here is that each cluster C_n does not actually need global information to calculate (13). Although it appears to require access to all $\tilde{\boldsymbol{\zeta}}_j$, $\forall j \in \mathcal{N}\backslash\{n\}$, one can readily observe that

$$\| \boldsymbol{\zeta}_n + \sum_{j\in\mathcal{N}}^{j\neq n} \tilde{\boldsymbol{\zeta}}_j \|_2^2 = \sum_{i\in\mathcal{N}} \sum_{m\in\mathcal{M}_i} \Big(\zeta_{nim} + \sum_{j\in\mathcal{N}}^{j\neq n} \tilde{\zeta}_{jim} \Big)^2 \tag{14}$$

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 ζ_{nim}, $\forall\ U_{im} \notin \mathcal{B}_n$. Correspondingly, the summation terms in (14) 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 (10) 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 $\mathcal{B}_n$, i.e. $C_j \in \mathcal{C}_n$ if and only if $\mathcal{B}_j \bigcap \mathcal{B}_n \neq \emptyset$, where we define $\mathcal{C}_n \subseteq \mathcal{N}$ as the set of clusters that constitute the communication neighborhood of cluster C_n.

After calculating $\hat{\boldsymbol{\zeta}}_n^k$ according to (13), 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

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

where τ is a stepsize, the determination of which is critical to the convergence properties of the method, as will be explained later on. Subsequently, the dual update is performed according to

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

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 $\mathcal{I}_{nm} = \{C_j : U_{nm} \in \mathcal{B}_j, \forall j \in \mathcal{N}\}$. The ADAL algorithm applied on the multi-cluster relay beamforming problem is summarized in Alg. 2.

Each SINR constraint for $U_{nm}, \forall n \in \mathcal{N}, m \in \mathcal{M}_n$ contains variables from all clusters $C_j \in \mathcal{I}_{nm}$, which does not neccessarily mean that these clusters are within range to exchange messages with each other, as is required by the aforementioned analysis. This fact affects both the communication pattern needed to ensure that every C_j obtains the necessary data for its penalty term in (10) and the dual update procedure in (12). A practical solution would be to let the users perform the update of the corresponding dual variables and also act as message relays in the message exchange phase between coupled clusters.

According to the convergence analysis in [13], the stepsize τ must be determined according to $\tau \approx \frac{2}{\max_{nm} |\mathcal{I}_{nm}|}$, where $|\cdot|$ denotes the cardinality of a set. However here, the terms $|\mathcal{I}_{nm}|$ are not static and predetermined, since they are obviously influenced by the beamforming choices of the clusters, which in turn are dynamically changing throughout the iterative evolution of ADAL. A practical, heuristic solution for this would be to start the distributed algorithm with a rather conservative choice of τ and then slowly, as the iterations progress and the choices 'stabilize', change τ by monitoring the value of $\max_{nm} |\mathcal{I}_{nm}|$. More attractively, if we have some reasonable knowledge of the expected interference patterns in a given setup, we could define τ accordingly.

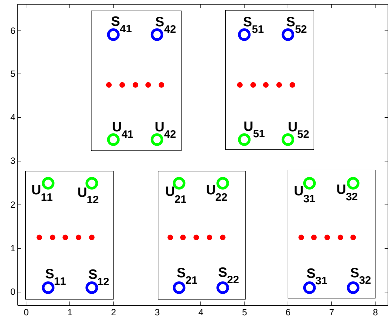


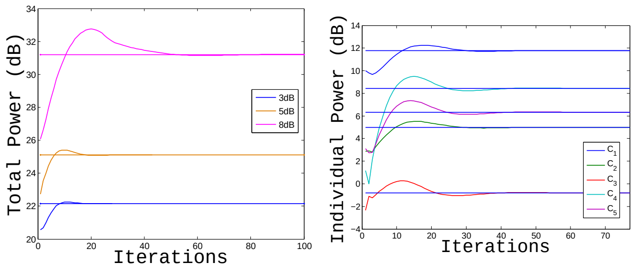
Fig. 2. Spatial configuration of a multi-cluster relay beamforming scenario with 5 clusters. The blue, green and red circles denote sources, destinations and relays, respectively.

IV. Numerical Analysis

In this section we illustrate the effectiveness of the proposed distributed algorithm under various settings. We follow a channel model encompassing large scale fading effects due to path loss and small scale fading. Specifically, we define the channel between two nodes k, l as $h_{kl} = \alpha_{kl} c_{kl} e^{j(2\pi/\lambda)d_{kl}}$, where α_{kl} captures multipath fading, $c_{kl} e^{j(2\pi/\lambda)d_{kl}}$ captures path loss, λ denotes the wavelength of carrier waves and d_{kl} denotes the Euclidean distance between nodes k and l. The path loss coefficient is a function of distance between the nodes given by $c_{kl} = d_{kl}^{-\mu/2}$, where $\mu = 3.4$ is the path loss exponent and represents the power fall-off rate. We do not include large-scale shadowing effects for simplicity, however the extension is possible. Also, we assume Rayleigh fading such that the gains $\alpha_{kl} \sim \mathcal{CN}(0, 1)$. Correspondingly, for the purpose of simulations we construct the various channel matrices of the general form $\mathbf{R}_{kl} = \mathbb{E}[\mathbf{h}_{kl}(t)\mathbf{h}_{kl}^{\mathcal{H}}(t)] \in \mathbb{C}^{N_T \times N_T}$ by generating realizations of Rayleigh random variables and setting $\mathbf{R}_{kl} = \mathbf{h}_{kl}\mathbf{h}_{kl}^{\mathcal{H}}$. The signal wavelength is 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).

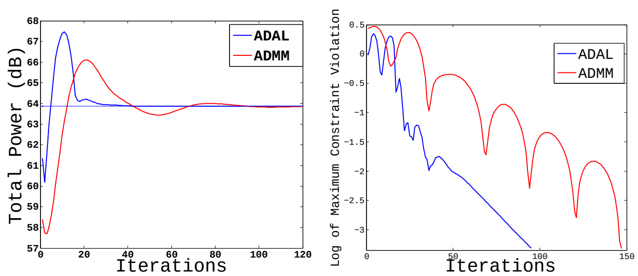
In all the cases presented below, we have always set the initial values of the primal and dual variables to zero in order to minimize the influence of initialization on the convergence behavior of the iterative distributed algorithm. The penalty parameter ρ is in general user defined in AL methods based on the effect it has on the behavior of each specific problem.

A typical spatial configuration of the networks considered in simulations is depicted in Fig. 2. All destinations are placed nearby such that the presence of interference renders the beamforming problems non-trivial. Fig. 3 presents the convergence results of ADAL on the scenario depicted in Fig. 2 and for various values of SINR requirements. In particular, Fig. 3(a) demonstrates convergence of the objective function (total transmitted power at all the relays of the system), while in Fig. 3(b) we plot the progress of the transmitted power at the relays of each cluster. In all cases, ADAL leads to very fast convergence. We must also mention that the entries of the



(a) (b)

Fig. 3. Convergence results of ADAL: a) Total transmitted power (at the relays) of the network scenario depicted in Fig. 2 for different SINR requirement values of γ. and b) Individual transmitted power (at the relays) of each cluster for the aforementioned setup and γ = 8dB. The straight lines in both figures indicate the respective centralized solutions.



(a) (b)

Fig. 4. Comparative convergence results for the ADAL and ADMM algorithms on a scenario with 12 clusters and γ = 10dB: a) Total transmitted power at the relays and b) Constraint feasibility for $\sum_{n\in\mathcal{N}} \boldsymbol{\zeta}_n = \mathbf{0}$.

beamforming matrices converge to the respective values of the centralized solution. Moreover, these convergence results indicate that scenarios with higher SINR requirement tend to converge slower. This is intuitive, since for higher values of γ the need for coordination between the beamformers increases, in an effort to satisfy the stricter QoS requirements.

In Fig. 4, we compare ADAL with another popular distributed AL method, namely the Alternating Directions Method of Multipliers (ADMM) [11, 17, 24]. We can observe that ADAL converges faster than ADMM, both in terms of objective function convergence and constraint feasibility.

V. Conclusions

In this paper, we considered the the problem of cooperative beamforming in relay networks, for scenarios where multiple clusters of source-destination node pairs, along with their dedicated relays, coexist in space. Utilizing the Semidefinite Relaxation technique, we formulated an approximation of the problem in convex programming form and proposed a novel, distributed optimization algorithm that allows for autonomous computation of the optimal beamforming decisions by each cluster. The proposed approach combines low computational complexity with the robustness and convergence speed properties of regularization, while at the same time it requires minimal communication overhead.

References

[1] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] H. Gharavi and K. Ban, "Multihop sensor network design for wideband communications," *Proc. IEEE*, vol. 91, no. 8, pp. 1221–1234, August 2003.

[7] H. Ochiai, P. Mitran, H. V. Poor, and V. Tarokh, "Collaborative beamforming for distributed wireless ad hoc sensor networks," *IEEE Trans. Signal Process*, vol. 53, no. 11, pp. 4110–4124, Nov. 2005.

[8] R. Mudumbai, U. Madhow, D. R. Brown, and H. V. Poor, "Distributed transmit beamforming: Challenges and recent progress," *IEEE Commun. Mag*, vol. 47, no. 2, pp. 102–110, Feb. 2009.

[9] A. Ephremides, "Energy concerns in wireless networks," *IEEE Trans. on Wireless Communications*, vol. 9, no. 4, pp. 48–59, August 2002.

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

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

[12] A. Nedic and A. E. Ozdaglar, "Subgradient methods in network resource allocation: Rate analysis," in *Proc. of CISS*, 2008, pp. 1189–1194.

[13] N. Chatzipanagiotis, D. Dentcheva, and M. Zavlanos, "Accelerated augmented Lagrangian decomposition," *Mathematical Programming*, 2013, (Submitted).

[14] P. Tatjewski, "New dual-type decomposition algorithm for nonconvex separable optimization problems," *Automatica*, vol. 25, no. 2, pp. 233–242, 1989.

[15] N. Watanabe, Y. Nishimura, and M. Matsubara, "Decomposition in large system optimization using the method of multipliers," *Journal of Optimization Theory and Applications*, vol. 25, no. 2, pp. 181–193, 1978.

[16] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximations," *Computers and Mathematics with Applications*, vol. 2, pp. 17–40, 1976.

[17] J. Eckstein and D. P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming,*, vol. 55, pp. 293–318, 1992.

[18] A. Ruszczynski, "On convergence of an Augmented Lagrangian decomposition method for sparse convex optimization," *Mathematics of Operations Research*, vol. 20, pp. 634–656, 1995.

[19] 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.

[20] 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.

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

[22] 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.

[23] R. Gallager, *Principles of Digital Communication*. Cambridge Univ. Press, Cambridge, UK, 2008.

[24] 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.