Distributed Multi-Stream Beamforming in MIMO Multi-Relay Interference Networks

CENK M. YETIS [ID], (Member, IEEE), AND RONALD Y. CHANG [ID], (Member, IEEE)
Research Center for Information Technology Innovation, Academia Sinica, Taipei 115, Taiwan

Corresponding author: Ronald Y. Chang (rchang@citi.sinica.edu.tw)

ABSTRACT In this paper, multi-stream transmission in interference networks aided by multiple amplify-and-forward (AF) relays in the presence of direct links is considered. The objective is to minimize the sum power of transmitters and relays by beamforming optimization under the stream signal-to-interference-plus-noise-ratio (SINR) constraints. For transmit beamforming optimization, the problem is a well-known non-convex quadratically constrained quadratic program (QCQP) that is NP-hard to solve. After semi-definite relaxation (SDR), the problem can be optimally solved via alternating direction method of multipliers (ADMM) algorithm for distributed implementation. Analytical and extensive numerical analyses demonstrate that the proposed ADMM solution converges to the optimal centralized solution. The convergence rate, computational complexity, and message exchange load of the proposed algorithm outperforms the existing solutions. Furthermore, by SINR approximation at the relay side, distributed joint transmit and relay beamforming optimization is also proposed that further improves the total power saving at the cost of increased complexity.

INDEX TERMS Alternating direction method of multipliers (ADMM), distributed multi-stream beamforming, MIMO multi-relay interference networks with direct links, quality of service assurance.

I. INTRODUCTION

The advent of future wireless networks bearing new components in large numbers including eNodeBs, cloud servers, relays, smart grids, massive multiple-input multiple-output (MIMO), and big data nodes, and the recent advances both in software and hardware architectures surging the applicability of parallel and distributed computations [1] have brought innovative solutions and paradigm shifts in recent years [2, Ch. 10] [3]–[5]. However, the distributed solutions based on conventional dual decomposition and other methods lack the effectiveness on the numerical stability, fast convergence rates [6], and scalability to high dimensional problems [7] compared to alternating direction method of multipliers (ADMM) which combines the strengths of dual decomposition and augmented Lagrangian methods [8].

ADMM studies in relay [9]–[11] and point-to-point [7], [12]–[16] networks are limited. In [9], an improved version of ADMM is proposed for power minimization under SINR constraints in multi-cluster relay networks with single antenna nodes. In [10], max-min SINR optimization is studied for decode-and-forward (DF) relay networks with single antenna transmitters and receivers under the constraints of total transmitter power, total relay power, and total number of relays. In [11], a distributed transmit power control algorithm via ADMM is proposed for a single relay aided network. ADMM in point-to-point networks is slightly more investigated in the literature. In [7], cloud radio access networks (C-RAN) with single antenna receivers are optimized for power minimization under SINR constraints. In [12], power minimization with the worst-case SINR constraints due to the channel state information (CSI) errors in multi-cell coordinated multiple-input single-output (MISO) networks is solved via semi-definite relaxation (SDR). In [13], power minimization problem with rate constraints in wireless sensors networks is studied. In [14], beamforming design for power minimization under SINR constraints is proposed in MISO downlink systems by solving second order cone problems (SOCP). In [15], max-min flow rate optimization problem is considered for software defined radio access networks (SD-RAN) with single antenna nodes under wired and wireless link constraints. In [16], a distributed power control algorithm is proposed for the utility maximization without SINR

constraints in interference networks with single antenna nodes.

The three prominent features of future high performance wireless networks are multi-stream transmissions, multi-antenna nodes, and the line-of-sights between transmitters and receivers. The existence of direct links between transmitters and receivers is an immediate outcome of deploying a large number of intermediate nodes in the network to close the distances between transmitters and receivers. In C-RANs, SD-RANs, and wireless relay networks, the wireless intermediate nodes are called radio access units [7], base stations [15], and relays, respectively. Wireless relay networks can be regarded as the wireless communications parts of C-RANs and SD-RANs, which embody wired communications parts in their architectures as well. Although the three mentioned features are critical in practical systems, many studies on relay networks are based on communications settings with lesser features due to the difficulties that arise from the coexistence of all three features [17]. Among many objective functions [18, Ch. 8], the mean square error (MSE) minimization problems are more tractable in relay networks with the three features. Nevertheless, the studies on MSE are still limited [19], and to the best of our knowledge, the MSE problem in a relay network with all the three features is studied only in [20].

The limited attributes of aforementioned researches on relay networks can diminish their applications in future practical systems. In this paper, we consider amplify-and-forward (AF) multi-relay interference networks with all the three mentioned attributes. The problem of interest in this network setting is to minimize the total power consumption of transmitters and relays with guaranteed quality of service (QoS) by distributed optimization of the transmit beamforming filters. The QoS metric chosen in this paper is stream SINR, which is interconnected to the data rate and bit error rate (BER) expressions. The problem is a member of non-convex quadratically constrained quadratic programming (QCQP) problems [21], [22], which is not directly amenable to distributed optimization due to the intricate stream SINR constraints. In particular, the SINR constraints are not in a linear form and also are not decoupled over streams for parallel and distributive implementation. We fit the problem into the ADMM framework, a potent tool for distributed optimization.

When designing a solution for the problem, the feasibility of problem, i.e., the SINR constraints (targets) must be jointly supported, must be assured in the first stage before solving the main problem in the second stage. There are two approaches to assure the success of the first stage, namely, deriving the feasibility conditions [23] and relaxing the initial conditions, e.g., searching for feasible SINR targets [4], [24] and reducing the number of users [25]. The derivation of the feasibility condition is challenging even in simpler networks. In [26], an approximate condition is derived for a multi-relay network with a single transmitter and a receiver. The feasibility search, on the other hand, can be as costly as the solving

the main problem in terms of the number of iterations and computational complexity per iteration.

In this paper, we propose to use random initializations of beamforming vectors to automatically determine feasible SINR targets with a high probability. All cases where the convergence is slow, fluctuant, and infeasible, i.e., the SINR targets are infeasible, make up less than 2% of the simulations presented in this paper. The elimination of these mentioned cases is beneficial in two important applications: (1) Accurate and extensive cross-analyses of crucial network parameters, and benchmarking with competitive schemes over these varying parameters can be executed in short times. (2) By weighting the auto assigned SINR targets with scalar variables, the feasibility search problem can be reduced to a simpler linear search problem as demonstrated in Section VII-F. The cross-analyses are accurate since no approximations are needed in contrast to the approximately derived feasibility conditions [26].

Relay beamforming design in the existence of direct links has been a long standing open problem due to the challenge in the expression of SINR metric in terms of relay filters as detailed in Section VI. In this paper, an SINR reformulation is proposed that gives good approximation when the direct channels and the effective channels between the transmitters and receivers are independent. The proposed distributed joint transmit and relay beamforming assures improved total power saving than that of only the distributed transmit beamforming optimization. However, since each relay serves all streams in the network, the complexity substantially increases.

The main contributions of this paper are summarized as follows:

- A generic network model with multiple multi-antenna nodes at all sides, i.e., the transmitter, relay, and receiver sides, to carry out multi-stream transmissions in the presence of direct links is transformed into a compact matrix system model.
- The proposed distributed ADMM algorithm achieves the optimal centralized solutions in the given generic network model. In addition, in terms of convergence rate, computational complexity, and message exchange load performance metrics, the proposed solution surpasses other optimal distributed algorithms.
- By eliminating the feasibility of problem stage automatically, an extensive evaluation of the effects of crucial network parameters on the system performance metrics are attained that reveals new insights in this paper.
- Analytical and numerical results demonstrate the lower computational complexity and message exchange load, higher convergence rate, i.e., lesser number of iterations, and finally the convergence of proposed distributed algorithm.
- SINR approximation at the relay side is proposed to implement distributed joint transmit and relay beamforming optimization that further improves the total power saving at the cost of increased complexity.
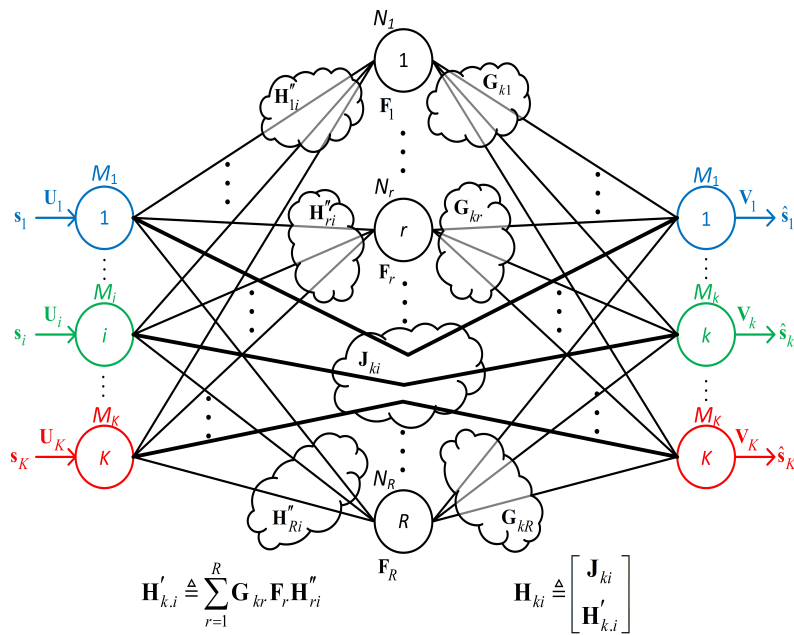
FIGURE 1. Multi-stream per user transmission in a K-user MIMO multi-relay interference network with direct links. Direct links are indicated by the bold lines.

The closest to our contribution in this paper is given in [11], where ADMM is also applied. The major difference between [11] and this paper is that a scalar power variable in a single relay network and a beamforming vector in a multi-relay aided network are optimized, respectively. Clearly, the extension in this paper is nontrivial. Further differences between [11] and this paper are discussed in later sections.

ADMM framework is applicable to many problems under mild conditions. The methodology is different than the distributed algorithms that are designed for particular problems. In [27], a distributed algorithm is proposed for interference alignment in signal space. To achieve the alignment in a distributed manner, each user minimizes the interference covariance matrix. Such distributed solutions that are designed for particular areas are in contrary to the distributed ADMM solutions that have broad application areas. In fact, ADMM solution for interference alignment is already exploited in [28].

The rest of the paper is organized as follows. The multi-stream transmission capable multi-relay interference network is introduced in Section II. In Section III, the transmit beamforming design problem for power minimization under stream SINR constraints is formulated. The distributed ADMM solution and benchmark distributed solutions are presented in Section IV. The attributes of distributed solutions including convergence, computational complexity, and message exchange load are studied in Section V. Distributed joint transmit and relay beamforming filter optimization via SINR approximation at the relay side is provided in Section VI. The numerical results and discussions are

presented in Section VII, and finally, the paper is concluded with the summary of main results in Section VIII.

II. SYSTEM MODEL

Consider a K-user two-hop MIMO multi-relay interference network aided by R relays. The ith source (transmitter) and the ith destination (receiver) each has M_i antennas while the rth relay has N_r antennas as shown in Fig. 1. Each transmitter communicates with its corresponding receiver with the aid of all relays. Without loss of generality, i^{th} transmitter and receiver pair can be called i^{th} user. We assume that all relay nodes work in half-duplex mode. Thus the communication between the users is completed in two time slots and there are non-negligible direct links between all transmitters and receivers. In the first time slot, the ith transmitter transmits the $M_i \times 1$ signal vector $\mathbf{x}_i = \mathbf{U}_i \mathbf{s}_i$, where $\mathbf{U}_i \in \mathbb{C}^{M_i \times d_i}$ and $\mathbf{s}_i \in \mathbb{C}^{d_i}$ are the transmit beamforming matrix and symbol vector with $E(\mathbf{s}_i \mathbf{s}_i^H) = \mathbf{I}_{d_i}$ and $E(\mathbf{s}_i \mathbf{s}_j^H) = 0$ for $j \neq i$, respectively. Here, d_i is the number of streams of the ith user, i.e., the number of independent data streams to be transmitted between the ith transmitter and receiver pair. We assume $d_i \leq M_i$ for sufficient degrees of freedom in signal detection. The transmitted signal from the user has a power constraint

$$p_i = E(||\mathbf{x}_i||^2) = \text{tr}(\mathbf{U}_i \mathbf{U}_i^H) \leq p_i^{\max}, \qquad (1)$$

where $p_i^{\max}$ is the maximum power of the ith transmitter. The received signal at the rth relay and k^{th} receiver in the first time slot is given by

$$\mathbf{y}_r = \sum_{i=1}^{K} \mathbf{H}''_{ri}\mathbf{x}_i + \mathbf{n}_r \quad \text{and} \quad \mathbf{y}_k(1) = \sum_{i=1}^{K} \mathbf{J}_{ki}\mathbf{x}_i + \mathbf{n}_k(1), \qquad (2)$$

respectively, where $\mathbf{H}''_{ri}$ is the channel from the ith transmitter to the rth relay, and $\mathbf{J}_{ki}$ is the channel between the ith transmitter and the kth receiver. $\mathbf{n}_r$ and $\mathbf{n}_k(1)$ are the complex additive white Gaussian noise (AWGN) at the rth relay and at the kth receiver in the first time slot with zero mean, and with the covariances $E(\mathbf{n}_r\mathbf{n}_r^H) = \sigma_r^2\mathbf{I}_{N_r}$ and $E(\mathbf{n}_k(1)\mathbf{n}_k^H(1)) = \sigma_k^2(1)\mathbf{I}_{M_k}$, respectively. In the second time slot, the received signal $\mathbf{y}_r$ is precoded at the rth relay by the $N_r \times N_r$ relay filter $\mathbf{F}_r$, $\mathbf{x}_r = \mathbf{F}_r\mathbf{y}_r$. The rth relay transmit power is $p_r = E(||\mathbf{x}_r||^2) = \mathrm{tr}(\mathbf{F}_r\mathbf{R}_{y_r}\mathbf{F}_r^H) \leq p_r^{\max}$, where $\mathbf{R}_{y_r} = E(\mathbf{y}_r\mathbf{y}_r^H)$ and $p_r^{\max}$ is the covariance matrix of the received signal at and the maximum power of the rth relay, respectively. The relay transmit power can be explicitly written as

$$p_r = \mathrm{tr}\left(\sum_{i=1}^{K}\mathbf{U}_i^H\mathbf{H}''^{H}_{ri}\mathbf{F}_r^H\mathbf{F}_r\mathbf{H}''_{ri}\mathbf{U}_i\right) + \sigma_r^2\mathrm{tr}\left(\mathbf{F}_r\mathbf{F}_r^H\right). \quad (3)$$

The received signal at the kth receiver in the second time slot is given by

$$\mathbf{y}_k(2) = \sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{x}_r + \mathbf{n}_k(2) = \sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{F}_r\mathbf{H}''_{rk}\mathbf{x}_k$$
$$+ \sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{F}_r\sum_{j=1,j\neq k}^{K}\mathbf{H}''_{rj}\mathbf{x}_j + \sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{F}_r\mathbf{n}_r + \mathbf{n}_k(2), \quad (4)$$

where $\mathbf{G}_{kr}$ is the channel between the rth relay and the kth receiver.

Define the effective channel from transmitter i to receiver k through all relays as

$$\mathbf{H}'_{k,i} \triangleq \sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{F}_r\mathbf{H}''_{ri}. \quad (5)$$

Then, $\mathbf{y}_k(2)$ in (4) is rewritten as

$$\mathbf{y}_k(2) = \mathbf{H}'_{k,k}\mathbf{x}_k + \sum_{j=1,j\neq k}^{K}\mathbf{H}'_{k,j}\mathbf{x}_j + \sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{F}_r\mathbf{n}_r + \mathbf{n}_k(2).$$

To obtain the SINR expression, the received signal at a receiver is written in terms of the desired signal, interference signal, and noise summands. Thus, define the aggregate channel matrix and noise vector as

$$\mathbf{H}_{ki} \triangleq \begin{bmatrix}\mathbf{J}_{ki} \\ \mathbf{H}'_{k,i}\end{bmatrix} \quad \text{and} \quad \mathbf{n}_k \triangleq \begin{bmatrix}\mathbf{n}_k(1) \\ \sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{F}_r\mathbf{n}_r + \mathbf{n}_k(2)\end{bmatrix}. \quad (6)$$

The list of channel notations used in the paper is given in Table 1 for convenience. Then, the aggregate received signal at receiver k can be written as

$$\mathbf{y}_k \triangleq \begin{bmatrix}\mathbf{y}_k(1) \\ \mathbf{y}_k(2)\end{bmatrix} = \mathbf{H}_{kk}\mathbf{U}_k\mathbf{s}_k + \sum_{j=1,j\neq k}^{K}\mathbf{H}_{kj}\mathbf{U}_j\mathbf{s}_j + \mathbf{n}_k. \quad (7)$$

For the sake of linear decoding complexity, the intra- and inter-user stream interferences are treated as noise in

TABLE 1. List of channel notations.

$\mathbf{J}_{ki}$	Channel from the ith transmitter to the kth receiver
$\mathbf{G}_{kr}$	Channel from the rth relay to the kth receiver
$\mathbf{H}''_{ri}$	Channel from the ith transmitter to the rth relay
$\mathbf{H}'_{k,i}$	Effective channel from the ith transmitter to the kth receiver through all relays
$\mathbf{H}_{ki}$	Aggregate effective channel from the ith transmitter to the kth receiver, i.e., stacked matrix of $\mathbf{J}_{ki}$ and $\mathbf{H}'_{k,i}$

this paper. The receive filter is $\bar{\mathbf{V}}_k \in \mathbb{C}^{2M_k \times d_k}$ since the receive filters for the time slots 1 and 2, $\mathbf{V}_k(1) \in \mathbb{C}^{M_k}$ and $\mathbf{V}_k(2) \in \mathbb{C}^{M_k}$, respectively, are stacked in this matrix, i.e., $\bar{\mathbf{V}}_k \triangleq \begin{bmatrix}\mathbf{V}_k(1) \\ \mathbf{V}_k(2)\end{bmatrix}$. After applying the receive filter $\bar{\mathbf{V}}_k$ to the received signal $\mathbf{y}_k$, the SINR of the lth stream of the kth user is obtained as

$$\mathrm{SINR}_{k,l} = \frac{\tilde{\xi}^{k,l}_{k,l}}{\sum_{(j,m)\neq(k,l)}\tilde{\xi}^{j,m}_{k,l} + \sigma^2_{n_{k,l}}}, \quad (8)$$

where $\tilde{\xi}^{i,n}_{k,l} \triangleq |\bar{\mathbf{v}}^H_{k,l}\mathbf{H}_{ki}\mathbf{u}_{i,n}|^2$ and $\bar{\mathbf{v}}_{k,l} \triangleq [\mathbf{v}^T_{k,l}(1) \, \mathbf{v}^T_{k,l}(2)]^T$ is the $2M_k \times 1$ receive beamforming vector for the lth stream of the kth user, i.e., l^{th} column of the receiver filter $\bar{\mathbf{V}}_k$ (similarly $\mathbf{v}_{k,l}(i), i = 1, 2$ are the lth columns of the receiver filters $\mathbf{V}_k(i)$, $i = 1, 2$, respectively), $\sigma^2_{n_{k,l}} = \bar{\mathbf{v}}^H_{k,l}\mathbf{R}_{n_k}\bar{\mathbf{v}}_{k,l}$ is the power of the aggregate noise after the receive beamforming

$$\mathbf{R}_{n_k} \triangleq \begin{pmatrix}\sigma_k^2(1)\mathbf{I}_{M_k} & \mathbf{0} \\ \mathbf{0} & \sum_{r=1}^{R}\sigma_r^2\mathbf{G}_{kr}\mathbf{F}_r\mathbf{F}_r^H\mathbf{G}_{kr}^H + \sigma_k^2(2)\mathbf{I}_{M_k}\end{pmatrix} \quad (9)$$

is the covariance matrix of the aggregate noise, and the notation $(j, m) \neq (k, l)$ denotes that $j \neq k$ and/or $m \neq l$.

III. PROBLEM FORMULATION

In this paper, the total power of transmitters and relays under SINR per stream constraints is minimized via the distributed ADMM algorithm [6] by optimizing all stream beamforming filters $\mathbf{u}_{k,l}, \forall k, l$ in parallel at each stream, aka processor, in the network. From (1) and (3), the total power, i.e., the sum of transmitter and relay powers, can be rewritten in terms of stream filters $\mathbf{u}_{k,l}$ (l^{th} column of the transmitter filter $\mathbf{U}_k$) as

$$\sum_{k=1}^{K}\sum_{l=1}^{d_k}\mathbf{u}^H_{k,l}\mathbf{R}_{.k}\mathbf{u}_{k,l}, \quad (10)$$

where $\mathbf{R}_{.k} \triangleq \mathbf{I}_{M_k} + \mathbf{R}'_{.k}$, $\mathbf{R}'_{.k} \triangleq \sum_{r=1}^{R}\mathbf{R}_{rk}$, and

$$\mathbf{R}_{rk} \triangleq \mathbf{H}''^{H}_{rk}\mathbf{F}_r^H\mathbf{F}_r\mathbf{H}''_{rk}. $$

Similarly, SINR of a stream can be rewritten as

$$\mathrm{SINR}_{k,l} = \frac{\mathbf{u}^H_{k,l}\mathbf{R}^k_{k,l}\mathbf{u}_{k,l}}{\sum_{(j,m)\neq(k,l)}\mathbf{u}^H_{j,m}\mathbf{R}^j_{k,l}\mathbf{u}_{j,m} + \sigma^2_{n_{k,l}}}, \quad (11)$$

where $\mathbf{R}^i_{k,l} \triangleq \mathbf{H}^H_{ki}\bar{\mathbf{v}}_{k,l}\bar{\mathbf{v}}^H_{k,l}\mathbf{H}_{ki}$. The superscript index in $\mathbf{R}^i_{k,l}$ indicates the ith transmitter, since $\mathbf{H}_{ki}$ is the aggregate channel between the ith transmitter and k^{th} receiver. The simplifications of $\mathbf{R}$ matrices are listed in Table 2 for convenience.

TABLE 2. List of some simplifications.

$\mathbf{R}_{rk}$	$\mathbf{H}_{rk}''^H \mathbf{F}_r^H \mathbf{F}_r \mathbf{H}_{rk}''$
$\mathbf{R}_{.k}'$	$\sum_{r=1}^{R} \mathbf{H}_{rk}''^H \mathbf{F}_r^H \mathbf{F}_r \mathbf{H}_{rk}''$
$\mathbf{R}_{.k}$	$\mathbf{I}_{M_k} + \mathbf{R}_{.k}'$
$\mathbf{R}_{k,l}^i$	$\mathbf{H}_{ki}^H \bar{\mathbf{v}}_{k,l} \bar{\mathbf{v}}_{k,l}^H \mathbf{H}_{ki}$

The problem formulation is given as
P1

$$\min_{\{\mathbf{u}_{k,l}\}\forall k\in\mathcal{K},\forall l\in\mathcal{L}_k} \sum_{k=1}^{K}\sum_{l=1}^{d_k} \mathbf{u}_{k,l}^H \mathbf{R}_{.k}\mathbf{u}_{k,l}$$

$$\text{s.t. } \text{SINR}_{k,l} \geq \gamma_{k,l}, \quad \forall k\in\mathcal{K}, \ \forall l\in\mathcal{L}_k \quad (12)$$

where $\mathcal{K} \triangleq \{1,2,\ldots,K\}$ and $\mathcal{L}_k \triangleq \{1,2,\ldots,d_k\}$ is the set of users and set of streams of the kth user, respectively.

IV. MULTI-STREAM BEAMFORMING UNDER STREAM SINR CONSTRAINTS

P1 is a non-convex quadratically constrained quadratic programming (QCQP) problem [21], [22]. To obtain the $\mathbf{u}_{k,l}$ filters, P1 can be equivalently rewritten by using the fact that $\mathbf{u}_{k,l}^H \mathbf{R}_{k,l}^i \mathbf{u}_{k,l} = \text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^i)$, where $\mathbf{X}_{k,l} \triangleq \mathbf{u}_{k,l}\mathbf{u}_{k,l}^H$, and by rewriting the SINR constraint as a summation inequality rather than a division inequality. P1 can be rewritten as
P2

$$\min_{\{\mathbf{X}_{k,l}\}\forall k,l} \sum_{k=1}^{K}\sum_{l=1}^{d_k} \text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k}) \quad (13a)$$

$$\text{s.t. } \frac{1}{\gamma_{k,l}}\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k) - \sum_{(j,m)\neq(k,l)} \text{tr}(\mathbf{X}_{j,m}\mathbf{R}_{k,l}^j) \geq \sigma_{n_{k,l}}^2,$$
$$\forall k\in\mathcal{K}, \ \forall l\in\mathcal{L}_k \quad (13b)$$

$$\mathbf{X}_{k,l} \in \mathbb{S}_+^{M_k}, \quad \forall k\in\mathcal{K}, \ \forall l\in\mathcal{L}_k \quad (13c)$$

$$\text{rank}(\mathbf{X}_{k,l}) = 1, \quad \forall k\in\mathcal{K}, \ \forall l\in\mathcal{L}_k. \quad (13d)$$

The transmit beamforming covariance matrix constraint (13c) imposes the convex constraint that $\mathbf{X}_{k,l}$ matrix belongs to the cone of symmetric and positive semi-definite matrices of dimension M_k (denoted by $\mathbb{S}_+^{M_k}$). Note that, since the covariance matrices in (13b), i.e., $\mathbf{R}_{k,l}^k$ and $\mathbf{R}_{k,l}^j$, are Hermitian, $\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^i) \in \mathbb{R}$, which means that the SINR inequality constraint (13b) is well defined. However, P2 is still non-convex due to the last constraint (13d), hence SDR can be applied, i.e., the last constraint can be relaxed [21], [22]. Nonetheless, the resulting SDR of P2 still cannot be solved distributively. To obtain $\mathbf{X}_{i,n}$ at the stream (i,n) via a parallel and distributed approach, both the objective function and constraints in P2 must be separable with respect to each stream. The objective function of P2 is separable; however, the reformulated SINR constraints (13b) are coupled. Moreover, since the SINR constraints are not linear, ADMM is not directly applicable to P2.

A. PROPOSED ADMM ALGORITHM

Before presenting the proposed ADMM algorithm, we briefly review the main steps of ADMM. ADMM can solve the following convex problem
P($\mathbf{x}, \mathbf{y}$)

$$\min_{\mathbf{x}\in\mathbb{C}^m, \mathbf{y}\in\mathbb{C}^n} f(\mathbf{x}) + g(\mathbf{y}) \quad (14a)$$

$$\text{s.t. } \mathbf{Ax} + \mathbf{By} = \mathbf{c} \quad (14b)$$

$$\mathbf{x} \in \mathcal{S}_1, \mathbf{y} \in \mathcal{S}_2, \quad (14c)$$

where $\mathbf{A} \in \mathbb{C}^{k\times m}$, $\mathbf{B} \in \mathbb{C}^{k\times n}$, $\mathbf{c} \in \mathbb{C}^k$, the functions f and g are convex, $\mathcal{S}_1$ and $\mathcal{S}_2$ are nonempty convex sets. Then the augmented Lagrangian for P($\mathbf{x}, \mathbf{y}$) is given as

$$\mathfrak{L}_\rho(\mathbf{x}, \mathbf{y}, \mathbf{z}) = f(\mathbf{x}) + g(\mathbf{y}) + \text{re}(\mathbf{z}^H(\mathbf{Ax}+\mathbf{By}-\mathbf{c}))$$
$$+ \frac{\rho}{2}\|\mathbf{Ax}+\mathbf{By}-\mathbf{c}\|_2^2, \quad (15)$$

where $\mathbf{z} \in \mathcal{C}^k$ is the Lagrange multiplier of the constraint (14b), re(.) is the real part operator, and ρ is again the Lagrangian dual update step size. P($\mathbf{x}, \mathbf{y}$) is solved by ADMM via three steps at each iteration s as follows

$$\mathbf{x}^{s+1} = \arg\min_{\mathbf{x}} \mathfrak{L}_\rho(\mathbf{x}, \mathbf{y}^s, \mathbf{z}^s) \quad (16a)$$

$$\mathbf{y}^{s+1} = \arg\min_{\mathbf{y}} \mathfrak{L}_\rho(\mathbf{x}^{s+1}, \mathbf{y}, \mathbf{z}^s) \quad (16b)$$

$$\mathbf{z}^{s+1} = \mathbf{z}^s + \rho(\mathbf{Ax}^{s+1} + \mathbf{By}^{s+1} - \mathbf{c}). \quad (16c)$$

In order to transform the reformulated SINR constraints to a linear form in P2, we initially introduce an auxiliary variable for each summand in the SINR constraint (13b) as

$$\zeta_{k,l} \triangleq \frac{1}{\gamma_{k,l}}\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k) - \sigma_{n_{k,l}}^2 \text{ and}$$

$$\zeta_{k,l}^{\backslash} \triangleq - \sum_{(j,m)\neq(k,l)} \text{tr}(\mathbf{X}_{j,m}\mathbf{R}_{k,l}^j). \quad (17)$$

Then, since the SINR constraints are active at the optimal point, i.e., the SINR constraints (13b) must hold with equality [29], the resulting SDR of P2 can be equivalently rewritten as
P3

$$\min_{\{\mathbf{X}_{k,l},\zeta_{k,l},\zeta_{k,l}^{\backslash}\}\forall k,l} \sum_{k=1}^{K}\sum_{l=1}^{d_k} \text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k}) \quad (18a)$$

$$\text{s.t. } \zeta_{k,l} + \zeta_{k,l}^{\backslash} = 0, \quad \forall k\in\mathcal{K}, \ \forall l\in\mathcal{L}_k \quad (18b)$$

$$\zeta_{k,l} = \frac{1}{\gamma_{k,l}}\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k) - \sigma_{n_{k,l}}^2, \quad \forall k\in\mathcal{K}, \ \forall l\in\mathcal{L}_k \quad (18c)$$

$$\zeta_{k,l}^{\backslash} = -\sum_{(j,m)\neq(k,l)} \text{tr}(\mathbf{X}_{j,m}\mathbf{R}_{k,l}^j), \quad \forall k\in\mathcal{K}, \ \forall l\in\mathcal{L}_k \quad (18d)$$

The constraint (13c).

Now, the SINR constraint (13b) is in a linear form via the constraints (18b), (18c), and (18d), and the coupling

constraint (18b) is a simple linear constraint that is viable for the ADMM algorithm. The partial augmented Lagrangian for P3 can be written as

$$
\mathcal{L}_\rho\big(\{\mathbf{X}_{k,l}, \zeta_{k,l}, \zeta^{\backprime}_{k,l}, \lambda_{k,l}\}_{\forall k,l}\big)
$$
$$
= \sum_{k=1}^{K} \sum_{l=1}^{d_k} \Big(\mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k}) + \lambda_{k,l}(\zeta_{k,l}+\zeta^{\backprime}_{k,l}) + \frac{\rho}{2}(\zeta_{k,l}+\zeta^{\backprime}_{k,l})^2 \Big),
$$
(19)

where $\lambda_{k,l}$ is the Lagrange multiplier of the constraint (18b), and $\rho \in \mathbb{R}_+$ is a positive constant parameter for adjusting the convergence speed, i.e., Lagrangian dual update step size.

P(**x**, **y**) and P3 have the following correspondences

$$
\mathbf{x} = [x_{11} \ldots x_{kl} \ldots x_{Kd_K}]^T, \quad \mathbf{y} = [\zeta_{k,l}\ \zeta^{\backprime}_{k,l}]^T
$$

$$
f(\mathbf{x}) = \sum_{k=1}^{K} \sum_{l=1}^{d_k} \mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k}), \quad g(\mathbf{y}) = 0
$$

$$
\mathbf{A} = 0, \quad \mathbf{B} = [1\ 1]^T, \quad \mathbf{c} = 0
$$

$$
\mathcal{S}_1 = \Big\{ \mathbf{x} | \mathbf{X}_{k,l} \in \mathbb{S}^M_+ \Big\},
$$

$$
\mathcal{S}_2 = \Big\{ \zeta_{k,l} \in \mathbb{R} | \zeta_{k,l} = \frac{1}{\gamma_{k,l}}\mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}^k_{k,l}) - \sigma^2_{n_{k,l}}, \forall k, l,
$$

$$
\zeta^{\backprime}_{k,l} \in \mathbb{R} | \zeta^{\backprime}_{k,l} = - \sum_{(j,m)\neq(k,l)} \mathrm{tr}(\mathbf{X}_{j,m}\mathbf{R}^j_{k,l}), \forall k, l \Big\},
$$
(20)

where $x_{kl} \triangleq f_x\big(\mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k})\big)$ and f_x is a mapping function between x_{kl} and $\mathbf{X}_{k,l}$.

As mentioned earlier, in [11], total power minimization under SINR constraints problem is solved via a distributed power control algorithm in a simpler network. In contrast, in this work, the problem is solved via beamforming vectors in a generic network. Hence, the problem in this work is more challenging and also the results are more effective than [11], i.e., higher sum-SINRs can be achieved with lesser power consumptions. Another distinction between [11] and this paper is the difference between the augmented Lagrangian functions, where both of the proposed solutions are fundamentally based on. As seen in [11, P2], the auxiliary definitions [11, eqs. (3c), (3d)] are augmented in the Lagrangian function [11, eq. (4)] to obtain closed-form solutions, whereas in this work, the summation of auxiliary terms (18b) are augmented as seen in (19). In [11], power control algorithm is proposed for a simpler relay network architecture. In other words, transmit power scalar variables $p_{k,l}$ are optimized in [11] as opposed to the transmit beamforming filters $\mathbf{u}_{k,l}$ in this work. The summation of auxiliary terms in [11, eq. (3b)] cannot be augmented in the Lagrangian function. Otherwise, the $p_{k,l}$ variable disappears in the Lagrangian differentiation process, hence closed-form solutions cannot be obtained. On the other hand, the auxiliary

definitions (18c) and (18d) cannot be augmented due to the trace operator in this work. Therefore, in this work, (18b) is augmented in the Lagrangian function and the covariance matrix solutions are obtained via CVX. Resorting to CVX for the solutions of covariance matrices is a common practice in the literature [9], [12]. After obtaining the transmit covariance matrices $\mathbf{X}_{k,l}$, obtaining the transmit vectors $\mathbf{u}_{k,l}$ is a well-known process [21], i.e., if the obtained covariance matrix is not rank-one, then additional rank-one approximate solution methods can be applied including the Gaussian randomization method. However, as observed by the numerical results in Section VII, the optimal $\mathbf{X}_{k,l}$ matrices are always rank-one. This indicates that the proposed distributed optimal solution to P3 serves as a global optimal solution to P1.

Remark 1: P3 is separable with respect to the streams, thus it requires solving B problems of P$\big(\mathbf{X}_{k,l}, \zeta_{k,l}, \zeta^{\backprime}_{k,l}\big)$, where $B \triangleq \sum_{i=1}^{K} d_i$ is the total number of streams in the network. However, in order to apply the two-block ADMM algorithm which is better understood than x-block ADMM algorithms in terms convergence [30], where $x > 2$, the variables $\big\{\mathbf{X}_{k,l}, \zeta_{k,l}, \zeta^{\backprime}_{k,l}\big\}_{\forall k\in\mathcal{K}, \forall l\in\mathcal{L}_k}$ can be divided into two groups $\{\mathbf{X}_{k,l}\}$ and $\big\{\zeta_{k,l}, \zeta^{\backprime}_{k,l}\big\}$. Hence, P3 is separated into two simpler parts: P$(\mathbf{X}_{k,l})$ and P$\big(\zeta_{k,l}, \zeta^{\backprime}_{k,l}\big)$. Therefore, ADMM can distributively solve P3 by solving $2B$ simpler subproblems in parallel.

THE MAIN STEPS OF ADMM
ADMM consists of sequential updates of primal variables $\mathbf{X}_{k,l}$, $\zeta_{k,l}$, and $\zeta^{\backprime}_{k,l}$, and the dual variables $\lambda_{k,l}$, $\mu_{k,l}$, and $\mu^{\backprime}_{k,l}$ [1] as presented below.

a: UPDATE OF $\mathbf{X}_{k,l}$
The smaller problem of $\mathbf{X}_{k,l}$ is
P$(\mathbf{X}_{k,l})$

$$
\min_{\mathbf{X}_{k,l}} \mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k})
$$
$$
\text{s.t. The constraints } (13c)' \text{ and } (18c)', \qquad (21)
$$

where the superscript $'$ denotes the constraints to be considered without the symbol $\forall$, e.g., $(18c)'$ denotes the constraint only for the stream (k, l), not for all streams $\forall k \in \mathcal{K}, \forall l \in \mathcal{L}_k$ as seen in (18c). The above linear problem can be solved locally at each processor (k, l) in parallel.

b: UPDATES OF $\zeta_{k,l}$ AND $\zeta^{\backprime}_{k,l}$
The smaller problem of $\zeta_{k,l}$ and $\zeta^{\backprime}_{k,l}$ is
P$(\zeta_{k,l}, \zeta^{\backprime}_{k,l})$

$$
\min_{\zeta_{k,l}, \zeta^{\backprime}_{k,l}} \lambda_{k,l}(\zeta_{k,l}+\zeta^{\backprime}_{k,l}) + \frac{\rho}{2}(\zeta_{k,l}+\zeta^{\backprime}_{k,l})^2
$$
$$
\text{s.t. The constraints } (18c)' \text{ and } (18d)'. \qquad (22)
$$

Then according to the Karush-Kuhn-Tucker (KKT) condition, the updates of auxiliary variables are given as

$$\zeta_{k,l}^{s+1} = -\frac{\lambda_{k,l}^s + \mu_{k,l}^s}{\rho} - \zeta_{k,l}^{\backslash s} \quad \text{and} \tag{23a}$$

$$\zeta_{k,l}^{\backslash s+1} = -\frac{\lambda_{k,l}^s + \mu_{k,l}^{\backslash s}}{\rho} - \zeta_{k,l}^{s+1}, \tag{23b}$$

where $\mu_{k,l}^s$ and $\mu_{k,l}^{\backslash s}$ are the Lagrangian multipliers of the constraints (18c) and (18d), respectively.

c: UPDATES OF $\lambda_{k,l}, \mu_{k,l}, \text{AND } \mu_{k,l}^{\backslash}$

The updates of dual variables $\lambda_{k,l}, \mu_{k,l}$, and $\mu_{k,l}^{\backslash}$ are given as

$$\lambda_{k,l}^{s+1} = \lambda_{k,l}^s + \rho\left(\zeta_{k,l}^{s+1} + \zeta_{k,l}^{\backslash s+1}\right), \tag{24a}$$

$$\mu_{k,l}^{s+1} = \mu_{k,l}^s + \rho_c\left(\zeta_{k,l}^{s+1} - \frac{1}{\gamma_{k,l}}\text{tr}\left(\mathbf{X}_{k,l}^{s+1}\mathbf{R}_{k,l}^k\right) + \sigma_{n_{k,l}}^2\right), \tag{24b}$$

$$\mu_{k,l}^{\backslash s+1} = \mu_{k,l}^{\backslash s} + \rho_c\left(\zeta_{k,l}^{\backslash s+1} + \sum_{(j,m)\neq(k,l)}\text{tr}\left(\mathbf{X}_{j,m}^{s+1}\mathbf{R}_{k,l}^j\right)\right), \tag{24c}$$

where $\rho_c \in \mathbb{R}_+$ is the conventional Lagrangian dual update step size.

B. PSEUDOCODE

The pseudocode of the proposed ADMM algorithm is given in Algorithm 1, where Matlab scripting language is used. As seen in step 7 of the algorithm, when the absolute deviation from the SINR target of each stream is within $\Delta_{k,l}^{\max}$ accuracy, the algorithm is terminated with a success flag.

C. FEASIBILITY OF PROBLEM

P3 can be infeasible for the given SINR targets $\gamma_{k,l}$. Therefore, the feasibility of P3 must be assured in the first stage before solving P3 in the second stage. There are basically two techniques in the literature: 1) the feasibility conditions are derived [23], and 2) the initial conditions are relaxed, e.g., the SINR targets are tested and the feasible targets are searched [4], [24]. Deriving the feasibility condition of P3 is challenging even for a simpler network and a simpler problem [26]. In [26], power minimization under the worst stream SINR condition is studied for a multi-antenna relay network with a single transmitter and a receiver, and the direct links are neglected. Furthermore, in [26], the feasibility condition is derived based on the signal-to-interference ratio (SIR) instead of SINR. After assuming the targets are feasible based on the approximate feasibility condition derived from the SIR metric, the beamforming vectors are solved based on SINR. The approximate feasibility condition is more accurate in the high SNR regime, where noise can be neglected. On the other hand, testing the SINR targets and searching for the feasible SINR targets [4], [24] are as costly as solving problem P3, i.e., both require high iteration

---

Algorithm 1 Proposed Distributed Multi-Stream Beamforming Algorithm via ADMM for MIMO Multi-Relay Interference Networks

    1) Randomly initialize Lagrangian variables, $\{\zeta_{k,l}^s, \zeta_{k,l}^{\backslash s}, \lambda_{k,l}^s, \mu_{k,l}^s, \mu_{k,l}^{\backslash s}\}\forall k,l$. 2) Initialize the stream powers equally, $\text{tr}(\mathbf{u}_{k,l}^s\mathbf{u}_{k,l}^{Hs}) = p_k^{\max}/d_k, \forall k, \forall l$, and relay powers, $p_r = p_r^{\max}, \forall r$. 3) Compute the stream SINRs, $\text{SINR}_{k,l}$, by randomly initializing the transmit and relay beamforming vectors, and assign the SINR target of a stream of user k to the average SINR of user k, $\gamma_k = \left(\sum_{l=1}^{d_k}\text{SINR}_{k,l}\right)/d_k$.

1: $iflag = 0$, % $iflag$: infeasibility flag
2: $s = 0$
3: **while** $s \leq s^{\max} - 1$ **do**
4:     Obtain $\mathbf{X}_{k,l}^{s+1}$ from P($\mathbf{X}_{k,l}$),$\forall k, \forall l$
5:     Obtain $(\zeta_{k,l}^{s+1}, \zeta_{k,l}^{\backslash s+1}), \lambda_{k,l}^{s+1}, \mu_{k,l}^{s+1}$, and $\mu_{k,l}^{\backslash s+1}, \forall k, \forall l$ from (23), (24a), (24b), and (24c), respectively
6:     $\Delta_{k,l}^{s+1} = |\text{SINR}_{k,l}^{s+1} - \gamma_k|, \forall k, \forall l$
7:     **if** $\Delta_{k,l}^{s+1} \leq \Delta_{k,l}^{\max}, \forall k, \forall l$ **then** $s = s^{\max}$, **end**
8:     $s = s + 1$
9: **end while**
10: **if** $p_k > p_k^{\max}$ or $p_r > p_r^{\max}, \forall k, r$ **then** $iflag = 1$, **end**

---

numbers and high computational complexities per iteration, which severely impedes the cross-analysis due to the long simulation durations.

In this work, we adopt a new technique. We randomly initialize the transmit and relay beamforming vectors with full powers, i.e., $p_k^{\max}, \forall k \in \mathcal{K}$ and $p_r^{\max}, \forall r \in \mathcal{R}$ to determine feasible SINR targets with a high probability. The infeasible cases that make up a small portion of tests along with slow and fluctuant converging cases are filtered through step 10 and the detection window as detailed earlier. Hence, by testing randomly and automatically generated feasible SINR targets for any permutation of the network parameters, the cross-analysis is executed systematically and extensively within short simulation durations. Since cross-analysis at this comprehensive level has not been performed in the literature yet, many new insights are revealed in this paper. For all simulations, the receive filters are also randomly initialized but normalized to unity since including the receiver power as another parameter in the already large set of cross-varying network parameters substantially complicates the cross-analysis in Section VII.

In fact, the transmit and relay power constraints

$$p_k = \sum_{l=1}^{d_k}\mathbf{u}_{k,l}^H\mathbf{u}_{k,l} = \sum_{l=1}^{d_k}\text{tr}(\mathbf{X}_{k,l}) \leq p_k^{\max} \quad \text{and}$$

$$p_r = \sum_{k=1}^{K}\sum_{l=1}^{d_k}\mathbf{u}_{k,l}^H\mathbf{R}_{rk}\mathbf{u}_{k,l} = \sum_{k=1}^{K}\sum_{l=1}^{d_k}\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{rk}) \leq p_r^{\prime\max}, \tag{25}$$

respectively, where $p_r^{\prime\max} \triangleq p_r^{\max} - \sigma_r^2\text{tr}(\mathbf{F}_r\mathbf{F}_r^H)$, can be incorporated into P($\mathbf{X}_{k,l}$). The Matlab script for CVX solution is given in Algorithm 2. However, instead of incorpo-

Algorithm 2 Matlab Script for Locally Solving P($\mathbf{X}_{k,l}$) With Incorporated Power Constraints at Each Stream in Parallel via CVX

cvx_begin sdp
    variable $\mathbf{X}_{k,l}(M,M)$ hermitian
    minimize(trace($\mathbf{X}_{k,l}\mathbf{R}_{.k}$));
    subject to
        trace($\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k$) == $\gamma_{k,l}(\zeta_{k,l} + \sigma_{n_{k,l}}^2)$;
        trace($\mathbf{X}_{k,l}$) <= $p_k^{\max} - \sum_{m=1,m\neq l}^{d_k}$ trace($\mathbf{X}_{k,m}$);
        for r=1:R
            trace($\mathbf{X}_{k,l}\mathbf{R}_{rk}$) <= $p_r'^{\max} - \sum_{(j,m)\neq(k,l)}$ trace($\mathbf{X}_{j,m}\mathbf{R}_{rj}$);
        end
        $\mathbf{X}_{k,l}$ >= 0;
cvx_end

rating the power constraints into CVX as shown in Algorithm 2, checking the power constraints at step 10 of Algorithm 1 has the following advantages. Firstly, the CVX algorithm runs faster without the incorporated power constraints, particularly for networks with high number of relays and relay antennas. Secondly, if the SINR targets are infeasible, the algorithm starts fluctuating, i.e., different SINRs that do not meet SINR targets are achieved over the iterations while the power constraints are still assured by the constraints incorporated in Algorithm 2. Therefore, it is time-consuming to detect whether the fluctuation is due to infeasibility or it is a rare case where the algorithm still converges after the fluctuation. These two cases significantly hinder the simulation durations and executing extensive cross-analysis in Section VII becomes impractical. As seen in Section VII, in the worst case, our proposed distributed algorithm requires 35 iterations on average. Therefore, instead of plugging the power constraints (25) into the CVX optimization, letting the algorithm converge to the SINR targets and then checking the power constraints by step 10 of Algorithm 1 can swiftly determine the feasibility of the targets.

D. BENCHMARK DISTRIBUTED ALGORITHMS

1) ADMM WITH BOUNDED GUARANTEE (ADMM-BG)

The conventional dual decomposition method [31] for a distributed solution of P3 is not applicable since the dual of P3 can be unbounded [12]. This can be demonstrated by considering to solve the dual problem
P($\mu_{k,l}, \mathbf{X}_{k,l}$)

$$\max_{\mu_{k,l}} \min_{\mathbf{X}_{k,l}} \mathcal{L}_{\rho_{k,l}}'^{\text{Conv.}}$$
$$\text{s.t. The constraint (13c)}', \qquad (26)$$

where

$$\mathcal{L}_{\rho_{k,l}}'^{\text{Conv.}}(\mathbf{X}_{k,l}, \eta_{k,l}') = \text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k}) - \eta_{k,l}'\frac{1}{\gamma_{k,l}}\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k)$$

instead of solving the subproblem P($\mathbf{X}_{k,l}$) in step-a of our proposed algorithm presented earlier via CVX.

The inner optimization problem of the above dual problem can be unbounded below given the dual variable $\mu_{k,l}$ so that $-\eta_{k,l}'\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k) \to -\infty$, i.e., the solution can go to minus infinity. This problem can be avoided by the extra quadratic penalty term added in the ADMM scheme

$$\mathcal{L}_{\rho_{k,l}}'^{\text{ADMM}}(\mathbf{X}_{k,l}, \eta_{k,l}', \zeta_{k,l})$$
$$= \text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k}) - \eta_{k,l}'\frac{1}{\gamma_{k,l}}\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k)$$
$$+ \frac{\rho}{2}\left(\zeta_{k,l} - \frac{1}{\gamma_{k,l}}\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k) + \sigma_{n_{k,l}}^2\right)^2. \qquad (27)$$

As seen in [12, eq. (23)], authors enforce the quadratic penalty term also for the first term $\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k})$ in (27) by defining an auxiliary variable

$$p_{k,l} \triangleq \text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k}), \quad \forall k \in \mathcal{K}, \ \forall l \in \mathcal{L}_k, \qquad (28)$$

and then introducing the slack variables $t_{k,l} \geq 0, \forall k \in \mathcal{K}, \forall l \in \mathcal{L}_k$ [12, eq. (29d)]

$$p_{k,l} = t_{k,l}. \qquad (29)$$

Note that the first term $\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{.k})$ in (27) is the objective term in P3, similar to [12, eq. (17a)]. Hence, the final local partial augmented Lagrangian function is given as

$$\mathcal{L}_{\rho_{k,l}}''^{\text{ADMM}}(\mathbf{X}_{k,l}, \eta_{k,l}'', \zeta_{k,l}, p_{k,l}, t_{k,l})$$
$$= p_{k,l} - \eta_{k,l}''p_{k,l}$$
$$+ \frac{\rho}{2}(t_{k,l} - p_{k,l})^2 - \eta_{k,l}'\frac{1}{\gamma_{k,l}}\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k)$$
$$+ \frac{\rho}{2}\left(\zeta_{k,l} - \frac{1}{\gamma_{k,l}}\text{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k) + \sigma_{n_{k,l}}^2\right)^2. \qquad (30)$$

Following the same principle from [12] as explained above, the modified local partial augmented Lagrangian function of (19) is

$$\mathcal{L}_{\rho_{k,l}}(\eta_{k,l}'', \zeta_{k,l}, \zeta_{k,l}`, p_{k,l}, t_{k,l}, \lambda_{k,l})$$
$$= p_{k,l} - \eta_{k,l}''p_{k,l}$$
$$+ \frac{\rho}{2}(t_{k,l} - p_{k,l})^2$$
$$+ \lambda_{k,l}(\zeta_{k,l} + \zeta_{k,l}`) + \frac{\rho}{2}(\zeta_{k,l} + \zeta_{k,l}`)^2, \qquad (31)$$

where $\lambda_{k,l}$ is the notation used in (19) that corresponds to $\eta_{k,l}'$ used in this subsection. Hence step-a of our proposed distributed algorithm is modified as follows
P($p_{k,l}, \mathbf{X}_{k,l}$)

$$\min_{p_{k,l}, \mathbf{X}_{k,l}} p_{k,l} - \eta_{k,l}''p_{k,l} + \frac{\rho}{2}(t_{k,l} - p_{k,l})^2$$
$$\text{s.t. The constraints (13c)}', (18c)', \text{ and } (28)', \qquad (32)$$

which can be again solved via CVX. Accordingly, the Algorithm 1 needs small modifications. In summary, in ADMM-BG, P($p_{k,l}, \mathbf{X}_{k,l}$) is solved instead of P($\mathbf{X}_{k,l}$) in step-a of our proposed algorithm to guarantee bounded solutions.

Due to the added superfluous variables and constraints in (28) and (29), the modified approach, ADMM-BG algorithm, has a slower convergence rate, i.e., more iterations are needed, and also has a higher computational complexity per iteration than the original approach in step-a, i.e., the proposed algorithm in this paper. As explained in more detail in Section V, as the problem size increases, i.e., more constraints in this case, the complexity of problem solution increases since CVX depends on the number of constraints. The iteration number also increases as a direct consequence of a larger problem size. Moreover, it is observed that ADMM-BG scheme is significantly more unstable, i.e., the rare frequencies of slow-, fluctuant-, and non-convergent cases significantly increase. Finally, more constraints directly necessitates more variables, i.e., more antennas at the transmitter side. Due to the constraint (28), the minimum number of transmitter antennas is more for ADMM-BG.

2) ACCELERATED DISTRIBUTED AUGMENTED LAGRANGIANS (ADAL)

In [9], an improved version of ADMM is proposed, which is coined as ADAL since it converges faster than the conventional ADMM. ADAL introduces an auxiliary variable for each interference term inside the summation [9, eq. (11), 3rd constraint] in contrast to the summation of interference terms (18d) as proposed in this work. Hence, instead of two constraints (18c) and (18d) per stream as proposed in this work, ADAL has 1 (desired signal) + $B - 1$ (interference signals) = B constraints per stream. More constraints increase 1) the iteration number (the problem size increases, similar to more number of users and antennas), 2) the computational complexity per iteration (the complexity of CVX is dependent on the number of constraints), and 3) the minimum number of antennas (the transmit covariance is involved in more constraints, thus more antennas-variables are needed). Following the approach in [9], the resulting SDR of P2 at the optimal point can be equivalently rewritten as P4

$$\min_{\{\mathbf{X}_{k,l}, \boldsymbol{\zeta}_{k,l}\}_{\forall k,l}} \sum_{k=1}^{K} \sum_{l=1}^{d_k} \mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}_k) \tag{33a}$$

$$\text{s.t.} \sum_{k=1}^{K} \sum_{l=1}^{d_k} \boldsymbol{\zeta}_{k,l} = 0, \tag{33b}$$

$$\zeta_{k,l}^{k,l} = \frac{1}{\gamma_{k,l}} \mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}_{k,l}^k) - \sigma_{n_{k,l}}^2, \quad \forall k \in \mathcal{K}, \ \forall l \in \mathcal{L}_k \tag{33c}$$

$$\zeta_{j,m}^{k,l} = -\mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}_{j,m}^k), \quad \forall k, j, \ j \neq k, \forall l, m, \ m \neq l \tag{33d}$$

The constraint (13c),

where $\boldsymbol{\zeta}_{k,l} \triangleq [\zeta_{1,1}^{k,l}, \ldots, \zeta_{i,n}^{k,l}, \ldots, \zeta_{K,d_K}^{k,l}]^T$ is the $B \times 1$ vector that stacks all auxiliaries $\zeta_{i,n}^{k,l}$, $\forall i \in \mathcal{K}, \forall n \in \mathcal{L}_i$. The ADAL-Direct algorithm proposed in [9] is shortly referred as ADAL in our work.

The partial augmented Lagrangian for P3 can be written as

$$\mathcal{L}_\rho\big(\{\mathbf{X}_{k,l}, \boldsymbol{\zeta}_{k,l}\}_{\forall k,l}, \boldsymbol{\lambda}\big)$$

$$= \sum_{k=1}^{K} \sum_{l=1}^{d_k} \mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}_k) + \boldsymbol{\lambda}^T \sum_{k=1}^{K} \sum_{l=1}^{d_k} \boldsymbol{\zeta}_{k,l}$$

$$+ \frac{\rho}{2} \left\| \sum_{k=1}^{K} \sum_{l=1}^{d_k} \boldsymbol{\zeta}_{k,l} \right\|_2^2, \tag{34}$$

where $\boldsymbol{\lambda} \triangleq [\lambda_{1,1}, \ldots, \lambda_{K,d_K}]^T$ is the vector of Lagrange multipliers of the constraint (18b), $\rho \in \mathbb{R}_+$ is the Lagrangian dual update step size, and $\|.\|_2$ is the L_2 vector norm operator. ADAL proposes to distribute the problem over streams as follows, i.e., the local partial augmented Lagrangian is given as

$$\mathcal{L}_{\rho_{k,l}}\big(\mathbf{X}_{k,l}, \boldsymbol{\zeta}_{k,l}, \vec{\boldsymbol{\zeta}}_{j,m}, \boldsymbol{\lambda}\big)$$

$$= \mathrm{tr}(\mathbf{X}_{k,l}\mathbf{R}_k) + \boldsymbol{\lambda}^T \boldsymbol{\zeta}_{k,l} + \frac{\rho}{2} \left\| \boldsymbol{\zeta}_{k,l} + \sum_{(j,m) \neq (k,l)} \vec{\boldsymbol{\zeta}}_{j,m} \right\|_2^2, \tag{35}$$

where $\vec{\boldsymbol{\zeta}}_{j,m}$ is the auxiliary variable transmitted from stream $(j,m), \forall j \neq k, \forall m \neq l$ to all streams $(i,n), \forall i \in \mathcal{K} \setminus \{j\}$, $\forall n \in \mathcal{L}_i \setminus \{m\}$. In other words, the $\vec{\boldsymbol{\zeta}}_{j,m}$ variable is evaluated at stream (j,m) and is transmitted to all streams $(i,n), \forall i \neq j, \forall n \neq m$, thus it is a constant vector for stream (k,l).

THE MAIN STEPS OF ADAL

a: UPDATES OF $\zeta_{k,l}$ AND $\mathbf{X}_{k,l}$

The smaller problem of $\boldsymbol{\zeta}_{k,l}$ and $\mathbf{X}_{k,l}$ is

$$P(\boldsymbol{\zeta}_{k,l}, \mathbf{X}_{k,l})$$

$$\min_{\boldsymbol{\zeta}_{k,l}, \mathbf{X}_{k,l}} \mathcal{L}_{\rho_{k,l}}$$

s.t. The constraints (13c)$'$, (33c)$'$, and (33d)$'$, (36)

where (33d)$'$ denotes the constraint (33d) without $\forall j, m$; that is, for a particular stream (k,l). For instance, assume $K = 3$ and $d = 2$, then for stream $(k,l) = (1,2)$, the set of ζ variables is $Z = \{\zeta_{1,1}^{1,2}, \zeta_{1,2}^{1,2}, \ldots, \zeta_{3,1}^{1,2}, \zeta_{3,2}^{1,2}\}$, where $|Z| = B$. The above linear problem can be solved locally at each processor in parallel via CVX in Matlab.

b: UPDATE OF $\vec{\zeta}_{k,l}$

The $\boldsymbol{\zeta}_{k,l}^s$ output of $P(\boldsymbol{\zeta}_{k,l}, \mathbf{X}_{k,l})$ is further updated as follows

$$\vec{\boldsymbol{\zeta}}_{k,l}^{s+1} = \vec{\boldsymbol{\zeta}}_{k,l}^s + \tau(\boldsymbol{\zeta}_{k,l}^s - \vec{\boldsymbol{\zeta}}_{k,l}^s), \tag{37}$$

where τ is a step size. $\vec{\boldsymbol{\zeta}}_{k,l}^{s+1}$ is to be transmitted to the other nodes as mentioned earlier.

c: UPDATE OF λ

As the final step, the dual variables are updated as follows

$$\boldsymbol{\lambda}^{s+1} = \boldsymbol{\lambda}^s + \tau\rho \sum_{k=1}^{K} \sum_{l=1}^{d_k} \vec{\boldsymbol{\zeta}}_{k,l}^{s+1}. \tag{38}$$

Note that the above update is also local to a processing node (k, l) since $\vec{\zeta}_{j,m}^{s+1}, \forall (j, m) \neq (k, l)$ updates are provided from other nodes than (k, l).

The $B-1$ constraints in (33d)$'$ of P($\boldsymbol{\zeta}_{k,l}, \mathbf{X}_{k,l}$) are overwhelming as opposed to our proposed ADMM algorithm, where the summation of these terms is utilized. Hence, the disadvantages of ADAL follow as mentioned earlier.

Remark 2: $\boldsymbol{\zeta}_{k,l}$ and $\mathbf{X}_{k,l}$ variables in P($\boldsymbol{\zeta}_{k,l}, \mathbf{X}_{k,l}$) are jointly solved via CVX. the closed-form solution of $\boldsymbol{\zeta}_{k,l}$ can be obtained from the KKT conditions, then only the $\mathbf{X}_{k,l}$ variables are solved via CVX to reduce the overall computational complexity.

Consider whether the joint solution of $\boldsymbol{\zeta}_{k,l}$ and $\mathbf{X}_{k,l}$ variables via CVX or closed-form solution of $\boldsymbol{\zeta}_{k,l}$ and CVX solution of $\mathbf{X}_{k,l}$ option should be chosen. As mentioned earlier, there are B equality constraints per stream due to the (33c)$'$ and (33d)$'$ constraints. With $\zeta_{k,l}^{i,n}$ (joint solution) each constraint is a bivariate equation of $\zeta_{k,l}^{i,n}$ (scalar) and $\mathbf{X}_{k,l}$ (matrix), whereas without $\zeta_{k,l}^{i,n}$ (no joint solution), each equality is a univariate equation of $\mathbf{X}_{k,l}$. Thus, in the univariate case, B is the minimum number of antennas required at a node, and in the bivariate case, the requirement is lower than B by the help of additional variables $\zeta_{k,l}^{i,n}$. As shown earlier, our proposed method has significantly lower number of equality constraints, thus both the iteration number (can be determined from numerical results since the computation of iteration number is not possible) and computational complexity (can be approximately computed in terms of limiting behavior) per iteration are significantly low. Therefore, we choose joint solution of $\boldsymbol{\zeta}_{k,l}$ and $\mathbf{X}_{k,l}$ via CVX for ADAL so that the iteration number comparison with our proposed distributed algorithm and the number of antennas in Section VII is more fair and convenient to compare.

From the above discussion, it can be also concluded that centralized algorithms can be less restrictive than the distributed algorithms in terms of minimum number of antenna requirements in general. The SDR of the QCQP problem P2 can be solved via CVX [32] in a centralized manner. Hence, for the centralized algorithm, there are Kd constraints and KM variables. On the other hand, for ADAL, there are $Kd + 1 \approx Kd$ constraints and M variables per processor. Therefore, the number of variables for the distributed algorithm is approximately shrunk $1/K$ times compared to the centralized solution. However, a centralized solution is less attractive when the dimension of the problem is large as mentioned earlier.

Similarly, it can be concluded that the power control algorithms are also more restrictive than the beamforming optimization problems. In particular, there is only 1 power variable $p_{k,l}$ per stream, on the other hand, there are M_k variables per stream in the transmit beamforming vector $\mathbf{u}_{k,l} \in \mathbb{C}^{M_k}$. Since the number of variables in the beamforming vectors that directly determine the SINR targets is significantly larger than the number of power variables, automatic generation of feasible SINR targets by random initializations of beamforming vectors is not possible for power control algorithms.

E. COMPETITIVE SINR TARGETS

P1, where the objective function is minimizing the total power and the QoS guarantee is meeting the SINR targets, is one of the most important optimization problems for resource allocation in wireless networks [7], [9], [11], [12], [14]. This problem can be considered as the dual interest of network utility maximization problems [33], [18, Ch. 8]. Depending on the system design goals, the trade-off between the consumed total power and the achieved sum-SINR in the network can be controlled by the given SINR targets.

The SINR targets can be determined at the physical or higher layers of the network. One of the approaches in the physical layer is to search for feasible SINR targets between the lower and upper bounds set by beamforming filters. For the lower bound, random filter initializations are good choices as supported by the numerical results in this paper. For the upper bound, SINR maximizing (max-SINR) filter initializations can be good choices as shown in [27]. In Section VII-F, we present the numerical results showing that by searching for higher feasible SINR targets between the lower and upper bounds, the system can achieve higher SINRs at the cost of increased power consumptions. In Appendix A, closed-form solutions of max-SINR filters for distributed implementation are provided.

V. ATTRIBUTES OF THE ALGORITHMS
A. CONVERGENCE
The global convergence of two-block ADMM is proven based on mild conditions in [30]. The optimal $\mathbf{X}_{k,l}$ solution for each subproblem of P3 is a global optimal solution for the complete SDR problem in P3 since at each iteration, each subproblem of P3 is convex, and the iterations are pursued until convergence. We analytically and numerically show in this section and in Section VII, respectively, that the proposed distributed algorithm is convergent.

The convergence of ADMM to the global optimum of P($\mathbf{x}, \mathbf{y}$) under mild conditions is given by the following lemma.

Lemma 1 [1, Proposition 4.2]: Assuming that the optimal solution set of P($\mathbf{x}, \mathbf{y}$) is nonempty, and $\mathbf{A}^T \mathbf{A}$ and $\mathbf{B}^T \mathbf{B}$ are invertible, the sequence of solutions $\{\mathbf{x}^{s+1}, \mathbf{y}^{s+1}, \mathbf{z}^{s+1}\}$ obtained from (16a), (16b), and (16c), respectively, is bounded and every limit point of $\{\mathbf{x}^{s+1}, \mathbf{y}^{s+1}\}$ is an optimal solution of P($\mathbf{x}, \mathbf{y}$).

Proposition 1: The iterates $\{\mathbf{X}_{k,l}^{s+1}, \zeta_{k,l}^{s+1}, \zeta_{k,l}^{\backslash s+1}\}_{\forall k,l}$ and $\{\lambda_{k,l}^{s+1}, \mu_{k,l}^{s+1}, \mu_{k,l}^{\backslash s+1}\}_{\forall k,l}$ in Algorithm 1 converge to the optimal solution of primal and dual problems as $s \to \infty$. When the algorithm converges, the optimal $\{\mathbf{X}_{k,l}\}_{\forall k,l}$ solutions obtained in step 4 of Algorithm 1 are the global optimal solutions of P3.

Proof: Since $g(\mathbf{y}) = 0$ and $\mathbf{A} = 0$, the Lagrangian function (15) is modified correspondingly, and also, the $\mathbf{A}^T \mathbf{A}$

condition in Lemma 1 is dropped. Based on Lemma 1 and the correspondences in (20), the proposition on the convergence of the proposed distributed algorithm follows. □

B. COMPUTATIONAL COMPLEXITY

The most computationally intensive part of the proposed ADMM algorithm is step 4 of Algorithm 1, where the general-rank positive semi-definite matrices $\{\mathbf{X}_{k,l}\}$ are obtained explicitly via CVX without closed-form solutions. Obtaining closed-form beamforming solutions is challenging even in simpler network architectures [9], [12]. After obtaining the optimal general-rank positive semi-definite matrices $\{\mathbf{X}_{k,l}\}$ via the proposed ADMM algorithm, rank-one solutions $\mathbf{u}_{k,l}$, where $\mathbf{u}_{k,l}\mathbf{u}_{k,l}^H = \mathbf{X}_{k,l}$, can be efficiently derived [21]. Although there are no closed-form solutions, semi-definite programming (SDP) has still many real-time applications [22], [34].

Remark 3: Similar to the subproblem of proposed distributed algorithm $P(\mathbf{X}_{k,l})$ in (21), note that the SDP solutions, e.g., via CVX, are also needed for the subproblems of benchmark schemes ADMM-BG $P(p_{k,l}, \mathbf{X}_{k,l})$ in (32) and ADAL $P(\boldsymbol{\zeta}_{k,l}, \mathbf{X}_{k,l})$ in (36).

The complexity of SDP solution for a generic SDR of a QCQP problem [21, eq. (5)] is given as $\mathcal{O}(\max\{M, c\}^4 M^{1/2} \log(1/\epsilon))$, where M is the antenna number, c is the number of constraints, and $\epsilon > 0$ is the solution accuracy. Note that $c = 1, 2,$ and B for the proposed algorithm, ADMM-BG, and ADAL, respectively. Hence, for the former two algorithms, $M \geq c$, while for the last algorithm, $M \geq c$ or $M < c$. For the sake of simplicity, we evaluate the complexity of algorithms based on $\mathcal{O}(Mc)$; that is, the limiting behavior of simple multiplication of the number of variables and constraints. The complexity benchmark of algorithms based on this simplified formula also matches with CPU time based simulation benchmarks as shown in the end of next section.

For the proposed distributed algorithm, $P(\mathbf{X}_{k,l})$ has M variables and 1 constraint, $P(\zeta_{k,l}, \zeta_{k,l}^{\backslash})$ has 2 variables that have closed-form solutions (23), and finally 3 more variables in (24), thus the complexity of proposed distributed algorithm is $\mathcal{O}(M + 5)$ per processor. For ADMM-BG, $P(p_{k,l}, \mathbf{X}_{k,l})$ has M (from $\mathbf{X}_{k,l}$) + 1 (from $p_{k,l}$) variables and 2 constraints. Due to the similar steps to our proposed algorithm, ADMM-BG also has $2 + 3 = 5$ variables, and additional 1 variable $t_{k,l}$. Hence, the complexity of ADMM-BG is $\mathcal{O}(2(M + 1) + 6)$. Finally, for ADAL, $P(\boldsymbol{\zeta}_{k,l}, \mathbf{X}_{k,l})$ has $M + B$ variables and B constraints, and B variables in each of (37) and (38). Thus, the complexity of ADAL is $\mathcal{O}(B(M + B) + 2B)$. In summary, the proposed algorithm has the lowest complexity, followed by ADMM-BG, and then ADAL. The numerical CPU benchmarks of the proposed and existing algorithms are presented in Section VII-D.

C. MESSAGE EXCHANGE LOAD

For the update of (24c) at each processor, a single value is needed, i.e., the summation term, which can be delivered by a collector node. This means that each processor needs to send out $B-1$ scalar values, one for each other processor, to the collector node. In total, $B(B-1)$ and B numbers of scalar values need to be exchanged from the processors to the collector node and from the collector node to the processors, respectively. As a result, in total, B^2 scalar values need to be exchanged in the network. Since the update (24c) is also needed for ADMM-BG, the message exchange loads of ADMM-BG and the proposed algorithm are same. For ADAL, the updates (38) are achieved by sending out (37) from each stream to the collector node. Therefore, for both message exchange directions between the collector node and processors, exchange of B^2 scalar values is needed, making a sum of $2B^2$ exchange of scalar values in the network. In summary, ADAL has the highest message exchange load, followed by ADMM-BG and the proposed algorithm.

Note that a deployed central node or any node among the existing nodes of relay network can serve as a collector node. Moreover, parallel updates of the variables in the ADMM algorithm are robust to delays and errors [6], [28].

VI. DISTRIBUTED JOINT TRANSMIT AND RELAY BEAMFORMING FILTER OPTIMIZATION

Relay filter design has been a long standing open problem when the direct links exist. To pinpoint the hurdle that direct links cause in relay filter design, SINR (8) is reformulated as an explicit function of relay filters next.

A. SINR REFORMULATION

Consider the signal power from stream n of transmitter i to stream l of receiver k

$$
\begin{aligned}
p_{klin} &= \mathbf{u}_{i,n}^H \left[\mathbf{J}_{ki}^H\ \mathbf{H}_{k.i}'^H \right] \begin{bmatrix} \mathbf{v}_{k,l}(1) \\ \mathbf{v}_{k,l}(2) \end{bmatrix} \left[\mathbf{v}_{k,l}^H(1)\mathbf{v}_{k,l}^H(2) \right] \begin{bmatrix} \mathbf{J}_{ki} \\ \mathbf{H}_{k.i}' \end{bmatrix} \mathbf{u}_{i,n} \\
&= \mathbf{u}_{i,n}^H \mathbf{J}_{ki}^H \mathbf{v}_{k,l}(1)\mathbf{v}_{k,l}^H(1)\mathbf{J}_{ki}\mathbf{u}_{i,n} \\
&\quad + \mathbf{u}_{i,n}^H \mathbf{H}_{k.i}'^H \mathbf{v}_{k,l}(2)\mathbf{v}_{k,l}^H(2)\mathbf{H}_{k.i}'\mathbf{u}_{i,n} \\
&\quad + \mathbf{u}_{i,n}^H (\mathbf{A}^H + \mathbf{A})\mathbf{u}_{i,n}, \quad\quad (39)
\end{aligned}
$$

where

$$
\mathbf{A} \triangleq \mathbf{J}_{ki}^H \mathbf{v}_{k,l}(1)\mathbf{v}_{k,l}^H(2)\mathbf{H}_{k.i}',
$$

$\mathbf{v}_{k,l}(1)$ and $\mathbf{v}_{k,l}(2)$ is the $M_k \times 1$ receive beamforming vector for the lth stream of the kth user in the first and second time slot, respectively.

The first and second summand in the last equality of (39) is independent of relay filters and can be rewritten in terms of Hermitian matrices. $\mathbf{A}^H + \mathbf{A}$ in the third summand is a Hermitian matrix, thus the third summand can be rewritten as $\mathrm{re}(2\mathbf{u}_{i,n}^H \mathbf{A}\mathbf{u}_{i,n})$, where

$$
\mathbf{u}_{i,n}^H \mathbf{A}\mathbf{u}_{i,n} = \mathbf{u}_{i,n}^H \mathbf{J}_{ki}^H \mathbf{v}_{k,l}(1)\mathbf{v}_{k,l}^H(2) \sum_{r=1}^{R} \mathbf{G}_{kr}\mathbf{F}_r\mathbf{H}_{ri}''\mathbf{u}_{i,n}. \quad (40)
$$

Hence, the signal power from stream n of transmitter i to stream l of receiver k via relay r is given as

$$
p_{klrin} = 2\mathrm{re}\left(\mathrm{tr}(\mathbf{Q}_{klrin}\mathbf{F}_r)\right), \quad (41)
$$

where

$$\mathbf{Q}_{klrin} \triangleq \mathbf{H}''_{ri}\mathbf{u}_{i,n}\mathbf{u}^H_{i,n}\mathbf{J}^H_{ki}\mathbf{v}_{k,l}(1)\mathbf{v}^H_{k,l}(2)\mathbf{G}_{kr}$$

is a non-Hermitian matrix. Due to the asymmetry that the direct links bring through the third summands of signal powers in (39), relay filter design in the existence of direct link is a non-trivial problem. The third summand is particularly not negligible when the direct channels $\mathbf{J}_{ki}$ and the effective channels $\mathbf{H}'_{k,i}$ are correlated. However, when these channels are independent, the affect of this term is less prominent. By omitting this term, the approximate SINR of the lth stream of the kth user is obtained as

$$\widehat{\mathrm{SINR}}_{k,l}(\mathbf{u}_{i,n}) = \frac{\zeta^{k,l}_{k,l}(1) + \zeta^{k,l}_{k,l}(2)}{\sum_{(j,m) \neq (k,l)}(\zeta^{j,m}_{k,l}(1) + \zeta^{j,m}_{k,l}(2)) + \sigma^2_{n_{k,l}}}, \tag{42}$$

where

$$\zeta^{i,n}_{k,l}(1) \triangleq \left|\mathbf{v}^H_{k,l}(1)\mathbf{J}_{ki}\mathbf{u}_{i,n}\right|^2,$$
$$\zeta^{i,n}_{k,l}(2) \triangleq \left|\mathbf{v}^H_{k,l}(2)\mathbf{H}'_{k,i}\mathbf{u}_{i,n}\right|^2, \quad \text{and}$$

$\sigma^2_{n_{k,l}}$ is the sum of noises in the first and second time slots that can be obtained from (9).

Next, (42) can be rewritten as an explicit function of relay filter

$$\widehat{\mathrm{SINR}}_{k,l}(\mathbf{f}_r) = \frac{\bar{\mathbf{f}}^H_r \tilde{\mathbf{C}}_{klrkl}\bar{\mathbf{f}}_r}{\bar{\mathbf{f}}^H_r \tilde{\mathbf{C}}_{klrjm}\bar{\mathbf{f}}_r} = \frac{\mathrm{tr}(\mathbf{Y}_r\tilde{\mathbf{C}}_{klrkl})}{\mathrm{tr}(\mathbf{Y}_r\tilde{\mathbf{C}}_{klrjm})}, \tag{43}$$

where $\mathbf{f}_r \triangleq \mathrm{vec}(\mathbf{F}_r)$, $\bar{\mathbf{f}}_r \triangleq \begin{bmatrix} \mathbf{f}_r \\ t \end{bmatrix}$, $|t|^2 = 1$, and $\mathbf{Y}_r \triangleq \bar{\mathbf{f}}_r\bar{\mathbf{f}}^H_r$. Further derivation details of (43) and the definition of $\tilde{\mathbf{C}}_{klrin}$ are given in Appendix B.

B. TOTAL POWER REFORMULATION

Thus far, the exact SINR (11) is approximated by (42), and the approximate SINR is rewritten as a function of relay filter in homogenous quadratic form (43). Next, the total power is rewritten as a function of relay filters.

Noting the relay power (3) and using the equality $\mathrm{tr}(\mathbf{ABA}^H) = \mathrm{vec}(\mathbf{A})^H(\mathbf{B} \otimes \mathbf{I})\mathrm{vec}(\mathbf{A})$, the total power is obtained as

$$\sum_{r=1}^{R} \mathbf{f}^H_r \mathbf{D}_{r.}\mathbf{f}_r, \tag{44}$$

where

$$\mathbf{D}_{r.} \triangleq \left(\sum_{k=1}^{K}\sum_{l=1}^{d_k}\mathbf{D}_{rkl} + \sigma^2_r\mathbf{I}_{N_r}\right)^T \otimes \mathbf{I}_N = \sum_{k=1}^{K}\sum_{l=1}^{d_k}\mathbf{D}'_{rkl},$$

$$\mathbf{D}_{rkl} \triangleq \mathbf{H}''_{rk}\mathbf{u}_{k,l}\mathbf{u}^H_{k,l}\mathbf{H}''^H_{rk}, \quad \text{and}$$

$$\mathbf{D}'_{rkl} \triangleq \left(\mathbf{D}_{rkl} + \frac{\sigma^2_r\mathbf{I}_{N_r}}{B}\right)^T \otimes \mathbf{I}_{N_r}.$$

Furthermore, (44) can be rewritten as

$$\sum_{r=1}^{R}\sum_{k=1}^{K}\sum_{l=1}^{d_k}\mathrm{tr}(\mathbf{Y}_r\bar{\mathbf{D}}'_{rkl}), \tag{45}$$

where

$$\bar{\mathbf{D}}'_{rkl} \triangleq \begin{bmatrix} \mathbf{D}'_{rkl} + (\rho_2/2)\mathbf{I}_{N^2_r} & \mathbf{0}_{N^2-1\times 1} \\ \mathbf{0}_{1\times N^2-1} & 0 \end{bmatrix}_{N^2+1\times N^2+1}.$$

C. PROBLEM FORMULATION

Using (43) and (45), the relay beamforming filter design problem is given as

P5

$$\min_{\{\mathbf{Y}_r\}_{\forall r}} \sum_{r=1}^{R}\sum_{k=1}^{K}\sum_{l=1}^{d_k}\mathrm{tr}(\mathbf{Y}_r\bar{\mathbf{D}}'_{rkl}) \tag{46a}$$

$$\text{s.t. } \dot{\zeta}^{k,l}_r + \dot{\zeta}^{k,l\backslash}_r = 0, \quad \forall k \in \mathcal{K}, \; \forall l \in \mathcal{L}_k, \; \forall r \in \mathcal{R} \tag{46b}$$

$$\dot{\zeta}^{k,l}_r = \frac{1}{\gamma_{k,l}}\mathrm{tr}(\mathbf{Y}_r\tilde{\mathbf{C}}_{klrkl}), \quad \forall k \in \mathcal{K}, \; \forall l \in \mathcal{L}_k, \; \forall r \in \mathcal{R} \tag{46c}$$

$$\dot{\zeta}^{k,l\backslash}_r = -\mathrm{tr}(\mathbf{Y}_r\tilde{\mathbf{C}}_{klrjm}), \quad \forall k \in \mathcal{K}, \; \forall l \in \mathcal{L}_k, \; \forall r \in \mathcal{R} \tag{46d}$$

$$\mathrm{tr}\left(\mathbf{Y}_r\sum_{k=1}^{K}\sum_{l=1}^{d_k}\bar{\mathbf{D}}'_{rkl}\right) \leq p^{\max}_r, \quad \forall r \in \mathcal{R}. \tag{46e}$$

$$\mathrm{tr}(\mathbf{Y}_r) = 1, \quad \forall r \in \mathcal{R} \tag{46f}$$

$$\mathbf{Y}_r \in \mathbb{S}^N_+, \quad \forall r \in \mathcal{R}. \tag{46g}$$

The partial augmented Lagrangian is given as

$$\mathcal{L}_{\dot{\rho}}\left(\{\mathbf{Y}_r, \dot{\zeta}^{k,l}_r, \dot{\zeta}^{k,l\backslash}_r, \dot{\lambda}^{k,l}_r\}_{\forall k,l,r}\right)$$
$$= \sum_{k=1}^{K}\sum_{l=1}^{d_k}\sum_{r=1}^{R}\left(\mathrm{tr}(\mathbf{Y}_r\bar{\mathbf{D}}'_{rkl}) + \dot{\lambda}^{k,l}_r(\dot{\zeta}^{k,l}_r + \dot{\zeta}^{k,l\backslash}_r)\right.$$
$$\left. + \frac{\dot{\rho}}{2}(\dot{\zeta}^{k,l}_r + \dot{\zeta}^{k,l\backslash}_r)^2\right), \tag{47}$$

where $\dot{\lambda}^{k,l}_r \in \mathbb{R}$ is the Lagrange multiplier of the constraint (46b), and $\dot{\rho} \in \mathbb{R}_+$ is the Lagrangian dual update step size. From hereafter, the distributed solution follows the similar steps of the distributed transmit beamforming filter design proposed in Section IV-A.

D. PSEUDOCODE

The algorithm for the distributed joint solution is obtained by adding two more steps after step 5 of Algorithm 1. Similar to the transmitter side optimization, the first step after step 5 of Algorithm 1 is obtaining $\{\mathbf{Y}_r\}$. The second step is obtaining the auxiliary variables $\{\dot{\zeta}^{k,l}_r, \dot{\zeta}^{k,l\backslash}_r\}$, and the Lagrangian multipliers $\{\dot{\mu}_{k,l}\}$ and $\{\dot{\mu}^{k,l\backslash}_{k,l}\}$ for the constraints (46c) and (46d), respectively.

The discussions of distributed transmit beamforming filter design hold for the distributed relay beamforming filter design as well. For instance, CVX optimization and closed-form solutions are utilized for $\{\mathbf{Y}_r\}$ and $\{\dot{\zeta}^{k,l}_r, \dot{\zeta}^{k,l\backslash}_r, \dot{\mu}_{k,l}, \dot{\mu}^{k,l\backslash}_{k,l}\}$, respectively, and the optimal $\mathbf{Y}_r$ matrices are observed to be always rank-one. However, in contrast to the transmit side optimization as discussed at the end of Section IV-C, the relay power constraint (46e) is better to be incorporated into the CVX optimization, and

the transmit power constraint (25) at the transmitter side as well, since the distributed joint algorithm is significantly more costly in computations due to the distributed relay filter optimization as shown in Section VII-G.

VII. NUMERICAL RESULTS

In this section, the performance of the proposed scheme is compared with the centralized, ADMM-BG, and ADAL solutions by cross-varying the network parameters. All distributed algorithms in this section are optimal, thus they achieve the same SINR targets and also have the same total power consumptions that are achieved by the centralized solution. The network parameters to be cross-varied are the number of transmit (M) and relay (N) antennas, the number of relays (R), and the transmit (SNR$_T$) and relay (SNR$_R$) SNRs.

A. SIMULATION SETTINGS

In our simulations, the distances between transmitter and relay sides, and relay and receiver sides are same and fixed to 1 km. The shadowing coefficient follows log-normal distribution with zero mean and standard deviation equal to 8. The power of receive filter $\bar{\mathbf{v}}_{k,l}$ is normalized to 1 by equal power allocation over the receive filters $\mathbf{v}_{k,l}(i)$, $i = 1, 2$, i.e., the receive filter power for each time slot is normalized to $\sqrt{0.5}$. The transmitters and relays have the same transmit power constraints among themselves, i.e., $p_k^{\max} = p_T^{\max}$, $\forall k \in \mathcal{K}$ and $p_r^{\max} = p_R^{\max}$, $\forall r \in \mathcal{R}$, where $\mathcal{R} \triangleq \{1, 2, \ldots, R\}$ is the set of relays, but $p_T^{\max}$ and $p_R^{\max}$ are not necessarily equal. Hence, the SNR at the transmitter and relay side can be defined as

$$\text{SNR}_T = \frac{p_T^{\max}}{\sigma^2} \quad \text{and} \quad \text{SNR}_R = \frac{p_R^{\max}}{\sigma^2}, \tag{48}$$

respectively, and $\sigma_k^2 = \sigma^2$, $\forall k \in \mathcal{K}$ is assumed for simplicity. Without loss of generality, the streams of a user are assumed to have the same SINR target, which is equal to the average SINR of the user, i.e., the average of user's stream SINRs. For both the proposed algorithm and ADMM-BG, the step sizes ρ and ρ_c are empirically tuned to 1.2 and 0.5, respectively. For ADAL, the step sizes ρ, ρ_c, and τ are tuned to 9, 0.5, and 0.3, respectively. The step sizes are kept constant for the simulations. If $|\text{SINR}_{k,l} - \gamma_{k,l}| \leq \Delta_{k,l}^{\max} = 10^{-4}$ is met by the stream, the stream is regarded as achieving the SINR target.

Finally, for the benchmarks of the proposed algorithm and centralized solution, 100 Monte Carlo channels are tested. Due to the slow convergence rates of ADMM-BG and ADAL, 20 Monte Carlo channels are tested for their benchmarks with the proposed algorithm.

B. SOME NOTES

For all simulations, random initializations of filters and channels are pre-saved in files to feed the same inputs to the algorithms. Thus, reliable conclusions with less number of channel tests can be reached. Especially for large network sizes, this approach significantly reduces the simulation durations. Therefore, the networks with many varying parameters

are tested in reasonable time durations by the use of pre-saved inputs.

The numerical results in this section are presented for $K = 3$ networks since cross-varying K as well significantly increases the data sets to be analyzed. In all tests, each user is assumed to have $d_k = 2$ streams. Thus, a total of $B = 6$ streams are transmitted in all network configurations. Inline with the objective function of P1, the total power curves are plotted in this section. On the other hand, although QoS assurance of P1 imposes an SINR target for each of the streams, a single curve, sum-SINR, is plotted instead of plotting 6 individual stream SINR curves for the sake of clarity of the figures.

For all simulations of the proposed distributed algorithm presented in this section excluding the Section VII-F, the percentage of all cases including slow-, fluctuant-, and non-convergent, i.e., infeasible SINR targets, makes up in total 1.93%. We note that in Section VI-F, we propose the linear search method to determine higher feasible SINR targets than those are set by random filter initializations. The next iteration of linear search method is proceeded depending on whether the result of the previous iteration is feasible or infeasible.

The conclusions drawn in this section by cross-varying these parameters are not straightforward, e.g., increasing R can assist in achieving the SINR targets while also increasing the power consumption. Moreover, as discussed later, the ratios of these parameters also significantly affect the outputs, e.g., the ratios of SNR$_T$ and SNR$_R$ have different effects on the outputs.

C. PROPOSED ADMM VS. CENTRALIZED SOLUTION (Figs. 2 – 6)
1) PERFORMANCES OVER NETWORK SIZES AND SNRS (Fig. 2)

In Figs. 2(a), 2(b), and 2(c), the numerical results of average total power, sum-SINR, and iteration number vs. M and R are plotted, respectively.

As seen in Fig. 2(a), the proposed algorithm achieves the optimal centralized algorithm solutions, i.e., they have the same power consumptions. As mentioned earlier, all algorithms achieve the given SINR targets as seen in Fig. 2(b), and the optimality of an algorithm is determined whether it has the same power consumption with the centralized solution or not. In all cases, the numerical sum-SINR results in Section VII equal to the sum of the given feasible SINR targets of each stream, which are determined by random initializations of beamforming vectors.

As seen in Figs. 2(a) and 2(b), the network K3-M4-N8-R9 has a higher sum-SINR and a lower total power consumption than K3-M3-N8-R10, respectively. Since the optimization problem, power minimization while achieving the SINR targets, is solved at the transmitter side, providing more resources, i.e., more antennas, to the transmitters returns better results, i.e., higher SINRs are achieved
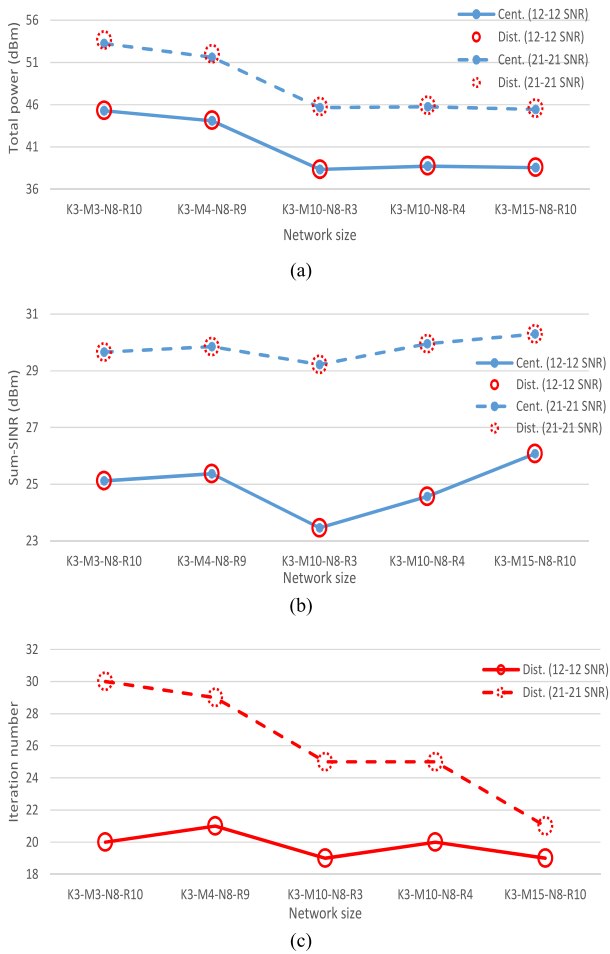
(a)

(b)

(c)

FIGURE 2. Average total power consumptions, sum-SINRs, and total number of iterations vs. network sizes and SNRs. (a) Average total power consumption vs. network sizes and SNRs. (b) Average sum-SINR vs. network sizes and SNRs. (c) Average total number of iterations of the proposed distributed algorithm.

with lower power consumptions. As expected, the sum-SINRs of the networks K3-M10-N8-R3, K3-M10-N8-R4, and K3-M15-N8-R10 gradually increase as the total numbers of relays in the networks increase. However, their total power consumptions remain similar because although R is increased, M is chosen sufficiently large for these networks.

As seen in Fig. 2(c), the iteration number increases when the SNR increases. When the transmitter and relay SNRs are 12 ($SNR_T = SNR_R = 12$), the iteration numbers are similar around 20 ± 1. Thus we focus on the $SNR_T = SNR_R = 21$ results. In our problem setting, where only transmit beamforming vectors are optimized, increasing M and decreasing R, decreases the iteration number. Lower R yields lower SINR, in other words, the range of numbers in consideration is lower, thus the iteration number is lower. Although higher M increases SINR, the iteration number decreases because, again, providing more resources, i.e., more antennas, to the transmitters returns better results,

i.e., less number of iterations. The algorithm converges faster because the degree of freedom is richer due to more resources, variables, available for the optimization of transmit beam-forming vectors.

2) PERFORMANCES OVER A WIDER RANGE OF SNRS FOR A PARTICULAR NETWORK SIZE (Fig. 3)

In Figs. 3(a), 3(b), and 3(c), the numerical results of average total power, sum-SINR and iteration number vs. transmit and relay SNRs for the K3-M10-N8-R3 network are presented, respectively. Again, the proposed algorithm achieves the optimal centralized solutions as seen in Fig. 3(a). In general, increasing SNR increases the power consumption
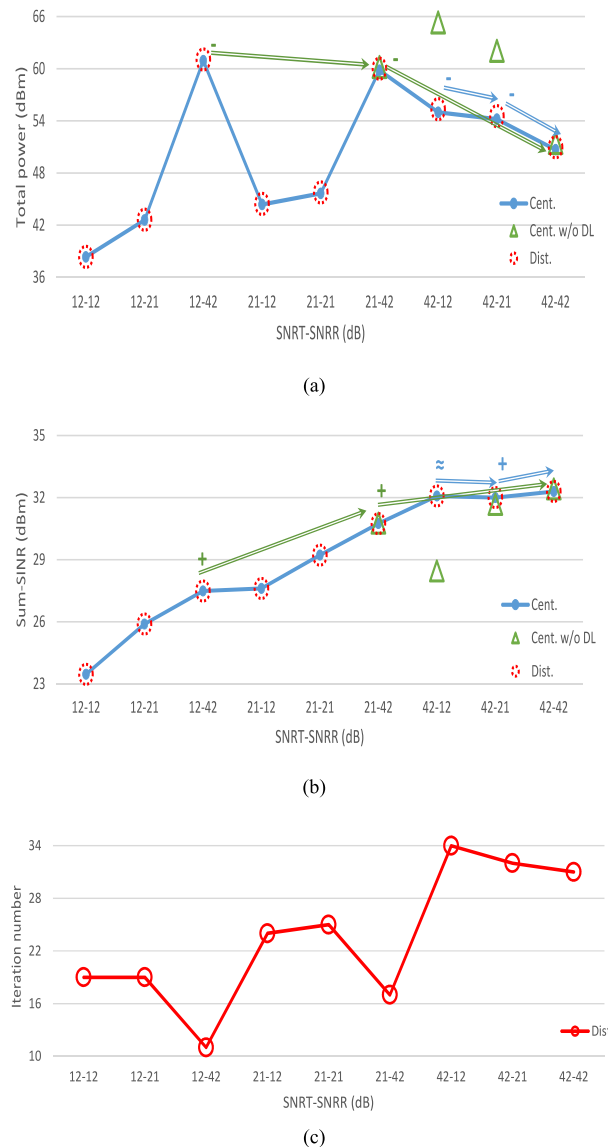

(a)

(b)

(c)

FIGURE 3. Average total power consumption, sum-SINR, and total number of iterations vs. SNRs in the K3-M10-N8-R3 network configuration. Green lines with arrows indicate the trends. (a) Average total power consumption vs. SNRs. (b) Average sum-SINR vs. SNRs. (c) Average total number of iterations vs. SNRs of the proposed distributed algorithm.

and also sum-SINR. However, consider the blue trend lines in Figs. 3(a) and 3(b). When SNR_T is sufficiently large, e.g., $SNR_T = 42$, increasing SNR_R can have a small effect on sum-SINR as seen in the 42-12, 42-21, and 42-42 dB results of Fig. 3(b). Next, consider the green trend lines in Figs. 3(a) and 3(b). When SNR_R is sufficiently large, increasing SNR_T is advantageous as seen in 12-42, 21-42, and 42-42 dB results of Figs. 3(a) and 3(b). $SNR_R = 42$ is sufficiently large to assist in achieving higher SINR targets, and providing more resources to the transmitter side returns better results, i.e., higher SINRs are achieved with lower power consumptions. On the other hand, $SNR_R = 12$ and 21 values are not sufficiently large. Thus, both the total power consumption and sum-SINR are increased by the increasing SNR_T, e.g., consider the 12-21, 21-21, and 42-21 dB results.

In Figs. 3(a) and 3(b), the numerical results for another benchmark that assumes the direct links between transmitters and receivers do not exist are presented. When SNR_T is high, more power is consumed while less sum-SINR is achieved as seen in the 42-12 and 42-21 dB results. However, when SNR_R is high, the effect of nonexisting direct links vanishes as seen in the 21-42 and 42-42 dB results.

As seen in Fig. 3(c), in general, the iteration number increases as SNR increases, by increasing both SNR_T and SNR_R, e.g., 12-12, 21-21, and 42-42 dB, and by only increasing SNR_T, e.g., 12-12, 21-12, and 42-12 dB. However, when only SNR_R is increased and reaches up to 42 dB, the iteration number decreases, i.e., compare 12-12 and 12-21 with 12-42, and compare 21-12 and 21-21 with 21-42 dB results. As seen in Fig. 3(a), there are notable peaks in power consumptions at the 12-42 and 21-42 dB points. The disproportionately high SNR_R value with respect to SNR_T helps in rapidly achieving the SINR targets, which cannot be controlled by the transmitter side, before the transmit beamforming optimization can further reduce the total power consumption. When the marginal $SNR_T = 42$ results are compared, as explained earlier, similar sum-SINR results are achieved for these dB points while total power consumptions are reduced from left to right as displayed in Fig. 3(a). As mentioned earlier, smaller range of numbers of interest, i.e., smaller power consumption values, decreases the iteration number. Thus the iteration number decreases gradually in the last three dB points of Fig. 3(c).

The total power savings can be obtained from the figures, i.e., total power saving (dB)$= (K p_T^{max} + R p_R^{max})$ (dB)$-$(y-axis value) (dB). As seen in the results, e.g., Figs. 2(a) and 3(a), the power savings are large as explained next. Due to the random initializations of beamforming vectors, the initial power budgets are likely to be highly redundant to achieve the SINR targets determined by these beamforming vectors. On the other hand, if max-SINR filters are used for the initializations instead of random initializations to determine the SINR targets, the SINR targets are likely to be infeasible.

3) SUPPORTING RESULTS FOR DISPROPORTIONATE SNRS OVER NETWORK SIZES (Fig. 4)

In Fig. 4, the early convergence of the algorithm when SNR_R is disproportionately high is shown again over different network sizes. Since the algorithm achieves the SINR targets rapidly due to the high SNR_R, the transmitter side lacks the opportunity to further reduce the power consumption similar to the case observed in Fig. 3.
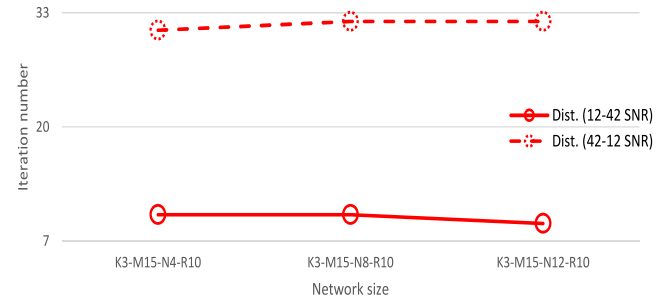

FIGURE 4. Average total number of iterations vs. network sizes and SNRs of the proposed distributed algorithm.

4) SCRUTINIZED PERFORMANCES OVER MULTIPLE AND SINGLE CHANNEL REALIZATIONS (Figs. 5 and 6)

In Fig. 5, total power consumptions vs. channel realizations are presented to demonstrate that the proposed algorithm achieves the optimal centralized solutions at each channel realization. In Figs. 6(a) and 6(b), the total power consumptions and total absolute target SINR deviations, $|\Delta| \triangleq |\sum_{k,l}^{K,d_k} \Delta_{k,l}|$ vs. iteration numbers are presented for a randomly selected channel realization.
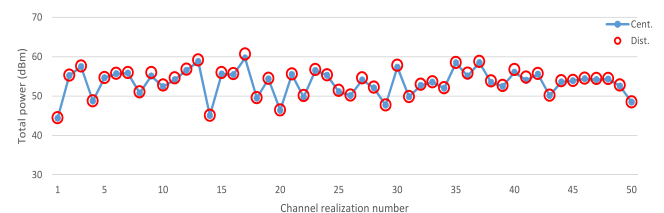

FIGURE 5. Total power consumption vs. channel realizations of the centralized and proposed distributed algorithms in the K3-M10-N8-R3 network configuration at $SNR_T = 42$ and $SNR_R = 12$ dB.

D. PROPOSED ADMM VS. EXISTING DISTRIBUTED ALGORITHMS (Table 3, Figs. 7 and 8)

The average CPU times for the aforementioned distributed algorithms in the K3-M6-N8-R14 network at $SNR_T = SNR_R = 21$ dB are shown in Table 3, based on a desktop computer with 64-bit operating system, Intel i7, CPU 3.40 GHz, and 16 GB RAM. A similar comparative trend is exhibited for other network configurations in our extensive experiments. The numerical results in Table 3 match with the analytical results obtained in Section V-B.
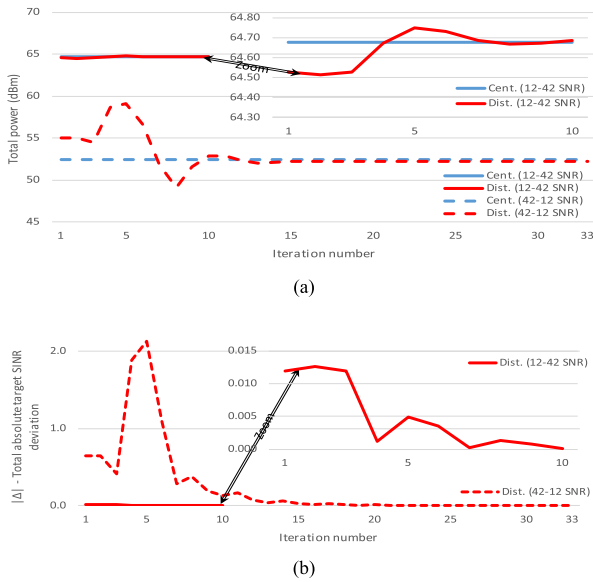
FIGURE 6. Total power consumption and absolute target SINR deviation vs. iteration and SNRs in the K3-M15-N8-R10 network configuration. (a) Total power consumption vs. iteration number. (b) Total absolute target SINR deviation vs. iteration number.

TABLE 3. Average CPU times of the proposed and distributed algorithms for the K3-M6-N8-R14 network.

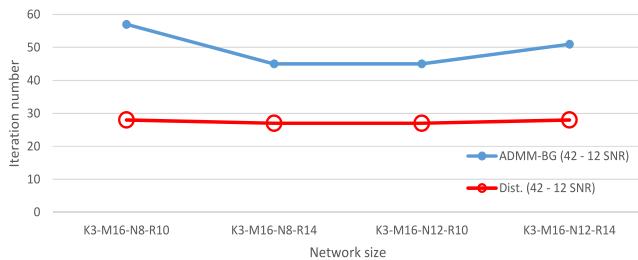| Algorithm	Proposed	ADMM-BG	ADAL
Average CPU time (sec.)	1.48	1.98	2.13

FIGURE 7. Average total number of iterations vs. network sizes.

As seen in Fig. 7, the iteration number of the proposed algorithm is significantly lower than ADMM-BG. The iteration numbers of the two algorithms are more distinct when disproportionate resources are allocated at the transmitter side, i.e., higher M and SNR_T. As expected, both algorithms achieve the SINR targets and the optimal centralized solutions, which are not plotted to avoid duplicate results.

In Fig. 8, the iteration numbers of the proposed and ADAL algorithms are compared, respectively. Each network configuration is tested at three different SNRs, particularly, from left to right at 12-12, 21-21, and 42-42 dBs. For the simplicity of figures, SNRs are not noted. As seen in Fig. 8, due to the significantly higher number of constraints in ADAL than the proposed algorithm, the iteration number of our proposed algorithm is always significantly lower than ADAL.
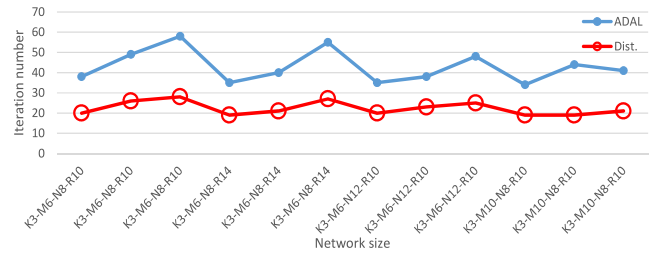

FIGURE 8. Average total number of iterations vs. network sizes and SNRs. Each network configuration is tested three times at three different SNRs.

E. BER PERFORMANCE OF PROPOSED ADMM (Fig. 9)

The important application areas of the SINR apparatus are deriving the closed-form BER and outage expressions [35], [36]. Therefore, as mentioned earlier, stream SINR is interconnected with the BER metric. Contrary to the rate results, the BER results cannot be obtained from the numerical SINR results presented in this section. In general, BER and outage performances improve as SINR improves, e.g., SINR outage is the probability of SINR descending below a preset SINR target. To illustrate this general trend, the sum-SINR and the corresponding BER improvement percentages of the system are demonstrated in Fig. 9. For instance, at $SNR_R = 21$ dB, BER is improved, i.e., decreased, by 1.59% by increasing the SNR_T from 6 to 12 dB, respectively. At $SNR_R = 12$ dB, 0.37% BER improvement is observed by increasing the SNR_T from 3 to 6 dB.
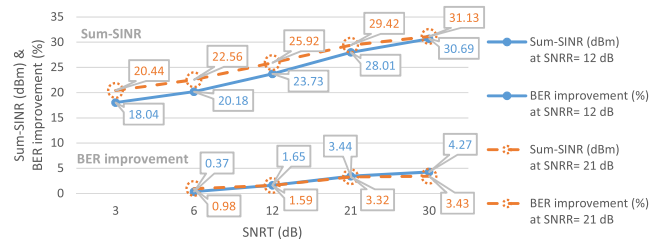

FIGURE 9. Average sum-SINR and the percentage of BER improvement vs. SNR_T of the proposed distributed algorithm in the K3-M10-N8-R3 network configuration.

The results in Fig. 9 are obtained over 300 Monte Carlo channels and uncoded QPSK modulation is used. For the earlier numerical results, the extraction of beamforming vectors $\mathbf{u}_{k,l}$ from $\mathbf{X}_{k,l}$ is not needed. However, for BER simulations, $\mathbf{u}_{k,l}$ needs to and can be extracted by rank-one decomposition [21] since the ranks of $\mathbf{X}_{k,l}$ matrices are observed to be always one [37], [38]. Based on this observation, the proposed SDR solution is also the global optimal solution to P1.

Contrarily, the maximum transmit powers SNR_T and SNR_R can be fixed and higher SINRs can still be achieved by searching for competitive SINR targets as demonstrated in the next section.

F. SEARCHING FOR COMPETITIVE SINR TARGETS (Fig. 10)

In this section, random filter and max-SINR filter initializations are considered to determine the lower and upper bounds of SINR targets, respectively. In Fig. 10, the leftmost bars indicate the results of the proposed distributed algorithm with random filter initializations, which is applied in all simulations of earlier sections. The rightmost bars present the results of distributed max-SINR algorithm. The bars on the left and right sides of the middle indicate the results of the proposed distributed algorithm with 1 iteration and 2 iterations of linear SINR target search, respectively. As the iteration number is increased, results closer to the SINRs achieved by the distributed max-SINR algorithm can be obtained. In summary, by fixing the transmit power constraints, but by searching for higher feasible SINR targets, higher SINRs can be achieved at the cost of increased power consumptions. This is an important design trade-off in wireless networks. The results in Fig. 10 are obtained over 20 Monte Carlo channels.
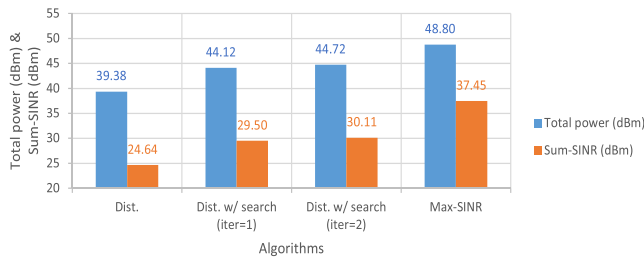

FIGURE 10. Average total power consumptions and sum-SINRs in the K3-M8-N8-R3 network configuration at 12-12 dB.

G. DISTRIBUTED JOINT TRANSMIT AND RELAY BEAMFORMING FILTER DESIGN (Table 4)

Distributed joint transmit and relay beamforming filter design improves the total power saving to achieve the same sum-SINR at the cost of increased complexity. Along the similar lines of the complexity analysis for transmitter side in Section V-B, the additional complexity due to the added distributed relay beamforming design is obtained as $\mathcal{O}\big(B(2N+5)\big)$ per relay, aka processor. Hence, as seen in Table 4 for the K3-M10-N10-R3 network, the additional total complexity for all relays per iteration is given by $\mathcal{O}\big(RB(2N+5)\big)$ or $\mathcal{O}\big(3\cdot6\cdot(2\cdot10+5)=450\big)$, where $B=6$ since each user transmits 2 streams as noted in the beginning of the section. On the other hand, as obtained in Section V-B, the total complexity of distributed transmitter optimization for all streams per iteration is $\mathcal{O}\big(B(M+5)=6\cdot(10+5)=90\big)$.

TABLE 4. Performances of distributed transmit vs. distributed joint transmit and relay beamforming filter designs for the K3-M10-N10-R3 network at 21-21 dB.

Algorithm Metric	Dist. $\{\mathbf{U}_k\}$	Dist. joint $\{\mathbf{U}_k, \mathbf{F}_r\}$
Average sum-SINR (dBm)	28.32	28.32
Average total power (dBm)	46.05	45.23
Average iteration number	16	23
Total complexity per iteration	$\mathcal{O}(90)$	$\mathcal{O}(90+450=540)$
Average total complexity	$\mathcal{O}(16\cdot90=1440)$	$\mathcal{O}(23\cdot540=12420)$

Each relay serves all streams to achieve the SINR targets. Thus, the distributed optimization at the relay side has higher number of constraints that substantially increases the total complexity. For the special case $M=N\gg5$, the ratio of the total complexities of the relay and transmitter sides is obtained as

$$\mathcal{O}\left(\frac{RB(2N+5)}{B(M+5)}\right) \approx \mathcal{O}(2R). \quad (49)$$

Due to the joint optimization, the average iteration number also increases as seen in Table 4. Therefore, the average total complexities of the distributed transmit and the distributed joint transmit and relay beamforming design algorithms are given as $\mathcal{O}(16\cdot90=1440)$ and $\mathcal{O}(23\cdot540=12420)$, respectively. The first three rows of results in Table 4 are obtained via numerical results averaged over 20 Monte Carlo channels. The total power saving increases as more resources are allocated to the relay side, i.e., $R>K$, $N>M$, and $\mathrm{SNR_R} > \mathrm{SNR_T}$.

The conclusions drawn from the numerical results of distributed transmit beamforming optimization are also valid for distributed joint transmit and relay beamforming optimization. For instance, recall the total power and sum-SINR results of increasing $\mathrm{SNR_T}$ 12-42, 21-42, 42-42 dB and increasing $\mathrm{SNR_R}$ 12-12, 12-21, 12-42 dB points in Fig. 3. For the former case, higher sum-SINRs are achieved with lower power consumptions. On the other hand, for the latter case, increasing $\mathrm{SNR_R}$ simply increases both the total power consumption and the sum-SINR. Clearly, for distributed transmit beamforming optimization, providing more resources to the transmitter side returns better results. On the other hand, for distributed joint transmit and relay beamforming optimization, providing more resources to the dominant side returns better results. For instance, when the relay side is dominant $R\gg K$, clearly, increasing $\mathrm{SNR_R}$ returns better results than increasing $\mathrm{SNR_T}$.

VIII. CONCLUSION

A distributed ADMM algorithm is proposed to design transmit beamforming matrices for a generic wireless relay network that has been hardly studied in the literature due to the challenges raised by the coexistence of multi-stream transmissions, multiple multi-antenna nodes, and the presence of direct links. The traits of the proposed algorithm are low complexity, iteration number, and message exchange loads. Due to the challenge that the direct links bring, an approximate SINR formulation at the relay side is proposed to design distributed joint transmit and relay beamforming filters that further improve the total power saving at the cost of increased complexity.

APPENDIX A
CLOSED-FORM
SOLUTIONS OF MAX-SINR FILTERS

Both downlink and uplink iterations of the distributed max-SINR algorithm are based on the covariance matrix of

interference signals plus noise, which is second-order statistics information. The covariance matrix of downlink direction is similar to the one given for interference channels, i.e., there are no relay nodes, in [27]. In downlink direction, based on the covariance matrix at stream (k, l)

$$
\mathbf{Q}_{k,l} = \sum_{j=1,j\neq k}^{K} \mathbf{H}_{kj}\mathbf{U}_{j}\mathbf{U}_{j}^{H}\mathbf{H}_{kj}^{H} + \sum_{m=1,m\neq l}^{d_k} \mathbf{H}_{kk}\mathbf{u}_{k,m}\mathbf{u}_{k,m}^{H}\mathbf{H}_{kk}^{H}
$$
$$
+ \mathbf{R}_{n_k}/d_k, \tag{50}
$$

the receive filter is obtained as

$$
\mathbf{v}_{k,l} = \frac{\mathbf{Q}_{k,l}^{-1}\mathbf{H}_{kk}\mathbf{u}_{k,l}}{\|\mathbf{Q}_{k,l}\|}. \tag{51}
$$

For the uplink iteration, we initially need to obtain the reciprocal channel $\overleftarrow{\mathbf{H}}'_{k,i}$ of $\mathbf{H}'_{k,i}$ in (5). The effective channel in the uplink direction from transmitter i to receiver k through all relays is given as

$$
\overleftarrow{\mathbf{H}}'_{k,i} = \sum_{r=1}^{R} \overleftarrow{\mathbf{H}}''_{kr}\overleftarrow{\mathbf{F}}_{r}\overleftarrow{\mathbf{G}}_{ri}, \tag{52}
$$

where $\overleftarrow{\mathbf{G}}_{ri} \triangleq \mathbf{G}_{ir}^{H}$ and $\overleftarrow{\mathbf{H}}''_{kr} \triangleq \mathbf{H}_{rk}^{H}$ are the $N_r \times M_i$ and $M_k \times N_r$ channel matrices between transmitter i and relay r, and relay r and receiver k, respectively. The covariance matrix of uplink direction can be obtained by interpreting the time domain, i.e., the two time slots due to the relay network communication, as the space domain. Thus, the covariance matrix at stream (k, l) is given as

$$
\overleftarrow{\mathbf{Q}}_{k,l} = \sum_{j=1,j\neq k}^{K} \left(\overleftarrow{\mathbf{J}}_{kj}\overleftarrow{\mathbf{U}}_{j}(1)\overleftarrow{\mathbf{U}}_{j}^{H}(1)\overleftarrow{\mathbf{J}}_{kj}^{H} \right.
$$
$$
\left. + \overleftarrow{\mathbf{H}}'_{k,j}\overleftarrow{\mathbf{U}}_{j}(2)\overleftarrow{\mathbf{U}}_{j}^{H}(2)\overleftarrow{\mathbf{H}}'^{H}_{k,j} \right)
$$
$$
+ \sum_{m=1,m\neq l}^{d_k} \left(\overleftarrow{\mathbf{J}}_{kk}\overleftarrow{\mathbf{u}}_{k,m}(1)\overleftarrow{\mathbf{u}}_{k,m}^{H}(1)\overleftarrow{\mathbf{J}}_{kk}^{H} \right.
$$
$$
\left. + \overleftarrow{\mathbf{H}}'_{k,k}\overleftarrow{\mathbf{u}}_{k,m}(2)\overleftarrow{\mathbf{u}}_{k,m}^{H}(2)\overleftarrow{\mathbf{H}}'^{H}_{k,k} \right) + \overleftarrow{\mathbf{R}}_{n_k}/d_k, \tag{53}
$$

where $\overleftarrow{\mathbf{J}}_{kj} \triangleq \mathbf{J}_{jk}^{H}$ is the $M_k \times M_j$ channel matrix for the direct link between transmitter j and receiver k, and

$$
\overleftarrow{\mathbf{R}}_{n_k} \triangleq \left(\sigma_k^2(1) + \sigma_k^2(2) \right)\mathbf{I}_{M_k} + \sigma_r^2 \sum_{r=1}^{R} \overleftarrow{\mathbf{H}}''_{kr}\overleftarrow{\mathbf{F}}_{r}\overleftarrow{\mathbf{F}}_{r}^{H}\overleftarrow{\mathbf{H}}''^{H}_{kr}
$$
$$
\tag{54}
$$

is the noise covariance matrix. Then,

$$
\overleftarrow{\mathbf{v}}'_{k,l} = \frac{\overleftarrow{\mathbf{Q}}_{k,l}^{-1}\left(\overleftarrow{\mathbf{J}}_{kk}\overleftarrow{\mathbf{u}}_{k,l}(1) + \overleftarrow{\mathbf{H}}'_{k,k}\overleftarrow{\mathbf{u}}_{k,l}(2) \right)}{\|\overleftarrow{\mathbf{Q}}_{k,l}\|} \tag{55}
$$

and finally, the receive filter is obtained as

$$
\overleftarrow{\mathbf{v}}_{k,l} = \sqrt{p_k/d_k}\frac{\overleftarrow{\mathbf{v}}'_{k,l}}{\|\overleftarrow{\mathbf{v}}'_{k,l}\|}. \tag{56}
$$

As detailed in [27], after obtaining the receive filter (51) in the downlink direction, the uplink direction is iterated by setting the transmit filter as $\overleftarrow{\mathbf{u}}_{k,l} = \mathbf{v}_{k,l}$. After obtaining the receive filter (56) in the uplink direction, the downlink direction is re-iterated by setting the transmit filter as $\mathbf{u}_{k,l} = \overleftarrow{\mathbf{v}}_{k,l}$. The uplink and downlink iterations are continued until the convergence.

APPENDIX B
DERIVATION DETAILS OF SINR APPROXIMATION (43)
The term $\zeta_{k,l}^{i,n}(2)$ in (42) can be rewritten as

$$
\mathbf{v}_{k,l}^{H}(2)\mathbf{H}'_{k,i}\mathbf{u}_{i,n} = \mathbf{v}_{k,l}^{H}(2)\sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{F}_{r}\mathbf{H}''_{ri}\mathbf{u}_{i,n}
$$
$$
= \sum_{r=1}^{R}\mathbf{a}_{klr}^{H}\mathbf{F}_{r}\mathbf{b}_{rin} = \mathbf{f}_{r}^{T}\mathbf{c}_{klrin} + r_{klsin}, \tag{57}
$$

where

$$
\mathbf{a}_{klr} \triangleq \mathbf{G}_{kr}^{H}\mathbf{v}_{k,l}(2), \quad \mathbf{b}_{rin} \triangleq \mathbf{H}''_{ri}\mathbf{u}_{i,n},
$$
$$
r_{klsin} \triangleq \sum_{s=1,s\neq r}^{R}\mathbf{a}_{kls}^{H}\mathbf{F}_{s}\mathbf{b}_{sin},
$$
$$
\mathbf{c}_{klrin} \triangleq \mathbf{b}_{rin} \otimes \mathbf{a}_{klr}^{*}, \quad \text{and } \mathbf{f}_{r} \triangleq \text{vec}(\mathbf{F}_{r}).
$$

$\otimes$ and vec(.) denotes the Kronecker product and denotes staking the columns of a matrix in a column vector, respectively. Therefore,

$$
\left|\mathbf{v}_{k,l}^{H}(2)\mathbf{H}'_{k,i}\mathbf{u}_{i,n}\right|^{2}
$$
$$
= (\mathbf{f}_{r}^{T}\mathbf{c}_{klrin} + r_{klsin})(\mathbf{f}_{r}^{T}\mathbf{c}_{klrin} + r_{klsin})^{*}
$$
$$
= \mathbf{f}_{r}^{T}\mathbf{c}_{klrin}\mathbf{f}_{r}^{H}\mathbf{c}_{klrin}^{*} + \mathbf{f}_{r}^{T}r_{klsin}^{*}\mathbf{c}_{klrin} + \mathbf{f}_{r}^{H}r_{klsin}\mathbf{c}_{klrin}^{*} + |r_{klsin}|^{2}
$$
$$
= \mathbf{f}_{r}^{H}\mathbf{c}_{klrin}^{*}\mathbf{c}_{klrin}^{T}\mathbf{f}_{r} + \mathbf{f}_{r}^{H}r_{klsin}\mathbf{c}_{klrin}^{*} + r_{klsin}^{*}\mathbf{c}_{klrin}^{T}\mathbf{f}_{r} + |r_{klsin}|^{2}
$$
$$
= \mathbf{f}_{r}^{H}\mathbf{C}_{klrin}\mathbf{f}_{r} + \mathbf{f}_{r}^{H}\mathbf{d}_{klrin} + \mathbf{d}_{klrin}^{H}\mathbf{f}_{r} + |r_{klsin}|^{2}, \tag{58}
$$

where

$$
\mathbf{C}_{klrin} \triangleq \mathbf{c}_{klrin}^{*}\mathbf{c}_{klrin}^{T} \quad \text{and } \mathbf{d}_{klrin} \triangleq r_{klsin}\mathbf{c}_{klrin}^{*}.
$$

Now consider the following term in (9)

$$
\mathbf{v}_{k,l}^{H}(2)\sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{F}_{r} = \sum_{r=1}^{R}\mathbf{a}_{klr}^{H}\mathbf{F}_{r}
$$
$$
= [\mathbf{f}_{r}^{T}\mathbf{o}_{klr1} \ \mathbf{f}_{r}^{T}\mathbf{o}_{klr2} \ldots \mathbf{f}_{r}^{T}\mathbf{o}_{klrN}] + \mathbf{t}_{kls}^{H}, \tag{59}
$$

where

$$
\mathbf{o}_{klrn} \triangleq \mathbf{e}_{n} \otimes \mathbf{a}_{klr}^{*}, \mathbf{t}_{kls} \triangleq \sum_{s=1 \ s\neq r}^{R}\mathbf{F}_{s}^{H}\mathbf{a}_{kls}, \quad \text{and}
$$

$\mathbf{e}_{n}$ is an $N \times 1$ zero vector except 1 at the nth row. Therefore,

$$
\left|\mathbf{v}_{k,l}^{H}(2)\sum_{r=1}^{R}\mathbf{G}_{kr}\mathbf{F}_{r}\right|^{2} = \sum_{n=1}^{N}\left|\mathbf{f}_{r}^{T}\mathbf{o}_{klrn}\right|^{2} + t_{kls}
$$
$$
= \mathbf{f}_{r}^{H}\left(\sum_{n=1}^{N}\mathbf{o}_{klrn}^{*}\mathbf{o}_{klrn}^{T}\right)\mathbf{f}_{r} + t_{kls}, \tag{60}
$$

where $t_{kls} \triangleq \mathbf{t}_{kls}^H \mathbf{t}_{kls}$. Hence, the noise power in the second time slot is

$$\sigma_{n_{k,l}}^2(2) = \sigma_r^2 \left| \mathbf{v}_{k,l}^H(2) \sum_{r=1}^R \mathbf{G}_{kr} \mathbf{F}_r \right|^2 + \sigma_k^2(2) \mathbf{v}_{k,l}^H(2) \mathbf{v}_{k,l}(2)$$
$$= \mathbf{f}_r^H \mathbf{O}_{klr} \mathbf{f}_r + \sigma_{klr}^2, \qquad (61)$$

where

$$\mathbf{O}_{klr} \triangleq \sigma_r^2 \sum_{n=1}^N \mathbf{o}_{klrn}^* \mathbf{o}_{klrn}^T \text{ and}$$
$$\sigma_{klr}^2 \triangleq \sigma_r^2 t_{kls} + \sigma_k^2(2) \mathbf{v}_{k,l}^H(2) \mathbf{v}_{k,l}(2).$$

Hence, the approximate SINR as a function of relay filter $\mathbf{f}_r$ is given as

$$\widehat{\text{SINR}}_{k,l}(\mathbf{f}_r) = \frac{x}{\sum_{(j,m)\neq(k,l)} (y) + \tilde{\sigma}_{n_{k,l}}^2}, \qquad (62)$$

where

$$x \triangleq \zeta_{k,l}^{k,l}(1) + \mathbf{f}_r^H \mathbf{C}_{klrkl} \mathbf{f}_r + \mathbf{f}_r^H \mathbf{d}_{klrkl} + \mathbf{d}_{klrkl}^H \mathbf{f}_r + |r_{klskl}|^2,$$
$$y \triangleq \zeta_{k,l}^{j,m}(1) + \mathbf{f}_r^H \mathbf{C}_{klrjm} \mathbf{f}_r + \mathbf{f}_r^H \mathbf{d}_{klrjm} + \mathbf{d}_{klrjm}^H \mathbf{f}_r + |r_{klsjm}|^2,$$
$$\tilde{\sigma}_{n_{k,l}}^2 \triangleq \sigma_k^2(1) \mathbf{v}_{k,l}^H(1) \mathbf{v}_{k,l}(1) + \mathbf{f}_r^H \mathbf{O}_{klr} \mathbf{f}_r + \sigma_{klr}^2.$$

Furthermore,

$$\widehat{\text{SINR}}_{k,l}(\mathbf{f}_r) = \frac{\mathbf{f}_r^H \mathbf{C}_{klrkl} \mathbf{f}_r + \mathbf{f}_r^H \mathbf{d}_{klrkl} + \mathbf{d}_{klrkl}^H \mathbf{f}_r + r_{1,klrkl}}{\mathbf{f}_r^H \bar{\mathbf{C}}_{klrjm} \mathbf{f}_r + \mathbf{f}_r^H \bar{\mathbf{d}}_{klrjm} + \bar{\mathbf{d}}_{klrjm}^H \mathbf{f}_r + r_{2,klrjm}}, \qquad (63)$$

where

$$r_{1,klrkl} \triangleq \zeta_{k,l}^{k,l}(1) + |r_{klskl}|^2,$$
$$\bar{\mathbf{C}}_{klrjm} \triangleq \sum_{(j,m)\neq(k,l)} \mathbf{C}_{klrjm} + \mathbf{O}_{klr},$$
$$\bar{\mathbf{d}}_{klrjm} \triangleq \sum_{(j,m)\neq(k,l)} \mathbf{d}_{klrjm} \text{ and}$$
$$r_{2,klrjm} \triangleq \sum_{(j,m)\neq(k,l)} \left(\zeta_{k,l}^{j,m}(1) + |r_{klsjm}|^2 \right)$$
$$+ \sigma_k^2(1) \mathbf{v}_{k,l}^H(1) \mathbf{v}_{k,l}(1) + \sigma_{klr}^2.$$

Note that $\widehat{\text{SINR}}_{k,l}(\mathbf{f}_r)$ is in an inhomogeneous quadratic form. To rewrite it in homogeneous quadratic form, let

$$\tilde{\mathbf{C}}_{klrkl} \triangleq \begin{bmatrix} \mathbf{C}_{klrkl} & \mathbf{d}_{klrkl} \\ \mathbf{d}_{klrkl}^H & r_{1,klrkl} \end{bmatrix} \text{ and}$$
$$\tilde{\mathbf{C}}_{klrjm} \triangleq \begin{bmatrix} \bar{\mathbf{C}}_{klrjm} & \bar{\mathbf{d}}_{klrjm} \\ \bar{\mathbf{d}}_{klrjm}^H & r_{2,klrjm} \end{bmatrix}. \qquad (64)$$

Thus, (43) is obtained.

REFERENCES

[1] D. P. Bertsekas, *Parallel and Distributed Computation: Numerical Methods*. Belmont, MA, USA: Athena Scientific, 2015.

[2] D. P. Palomar and Y. C. Eldar, *Convex Optimization for Signal Processing and Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2009.

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

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

[5] H. Pennanen, A. Tölli, J. Kaleva, P. Komulainen, and M. Latva-Aho, "Decentralized linear transceiver design and signaling strategies for sum power minimization in multi-cell MIMO systems," *IEEE Trans. Signal Process.*, vol. 64, no. 7, pp. 1729–1743, Apr. 2016.

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

[7] Y. Shi, J. Zhang, B. O'Donoghue, and K. B. Letaief, "Large-scale convex optimization for dense wireless cooperative networks," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4729–4743, Sep. 2015.

[8] Q. Ling, Y. Liu, W. Shi, and Z. Tian, "Weighted ADMM for fast decentralized network optimization," *IEEE Trans. Signal Process.*, vol. 64, no. 22, pp. 5930–5942, Nov. 2016.

[9] N. Chatzipanagiotis, Y. Liu, A. Petropulu, and M. M. Zavlanos, "Distributed cooperative beamforming in multi-source multi-destination clustered systems," *IEEE Trans. Signal Process.*, vol. 62, no. 23, pp. 6105–6117, Dec. 2014.

[10] J. Lin, Q. Li, C. Jiang, and H. Shao, "Joint multirelay selection, power allocation, and beamformer design for multiuser decode-and-forward relay networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 7, pp. 5073–5087, Jul. 2016.

[11] C. M. Yetis and R. Y. Chang, "Distributed power allocation in MIMO interference relay networks with direct links via ADMM," in *Proc. IEEE 86th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2017, pp. 1–5.

[12] C. Shen *et al.* "Distributed robust multicell coordinated beamforming with imperfect CSI: An ADMM approach," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2988–3003, Jun. 2012.

[13] M. Leinonen, M. Codreanu, and M. Juntti, "Distributed joint resource and routing optimization in wireless sensor networks via alternating direction method of multipliers," *IEEE Trans. Wireless Commun.*, vol. 12, no. 11, pp. 5454–5467, Nov. 2013.

[14] S. Joshi, M. Codreanu, and M. Latva-Aho, "Distributed resource allocation for MISO downlink systems via the alternating direction method of multipliers," in *Proc. Conf. Rec. 46th Asilomar Conf. Signals, Syst. Comput. (ASILOMAR)*, Nov. 2012, pp. 488–493.

[15] W.-C. Liao, M. Hong, H. Farmanbar, X. Li, Z.-Q. Luo, and H. Zhang, "Min flow rate maximization for software defined radio access networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1282–1294, Jun. 2014.

[16] S. Liao, J. Sun, Y. Chen, Y. Wang, and P. Zhang, "Distributed power control for wireless networks via the alternating direction method of multipliers," *J. Netw. Comput. Appl.*, vol. 55, pp. 81–88, Sep. 2015, doi: 10.1016/j.jnca.2015.05.005.

[17] D. Vouyioukas, "A survey on beamforming techniques for wireless MIMO relay networks," *Int. J. Antennas Propag.*, vol. 2013, Oct. 2013, Art. no. 745018, doi: 10.1155/2013/745018.

[18] N. D. Sidiropoulos, F. Gini, R. Chellappa, and S. Theodoridis, *Communications and Radar Signal Processing* (Academic Press Library in Signal Processing). Amsterdam, The Netherlands: Elsevier, 2014.

[19] K. X. Nguyen, Y. Rong, and S. Nordholm, "MMSE-based transceiver design algorithms for interference MIMO relay systems," *IEEE Trans. Wireless Commun.*, vol. 14, no. 11, pp. 6414–6424, Nov. 2015.

[20] K. X. Nguyen, Y. Rong, and S. Nordholm, "MMSE-based joint source and relay optimization for interference MIMO relay systems," *EURASIP J. Wireless Commun. Netw.*, vol. 2015, pp. 1–9, Mar. 2015.

[21] Z.-Q. Luo, W.-K. Ma, A. M.-C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.

[22] A. B. Gershman, N. D. Sidiropoulos, S. Shahbazpanahi, M. Bengtsson, and B. Ottersten, "Convex optimization-based beamforming," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 62–75, May 2010.

[23] A. Wiesel, Y. C. Eldar, and S. Shamai (Shitz), "Linear precoding via conic optimization for fixed MIMO receivers," *IEEE Trans. Signal Process.*, vol. 54, no. 1, pp. 161–176, Jan. 2006.

[24] M. Schubert and H. Boche, "Solution of the multiuser downlink beamforming problem with individual SINR constraints," *IEEE Trans. Veh. Technol.*, vol. 53, no. 1, pp. 18–28, Jan. 2004.

[25] K. T. Phan, T. Le-Ngoc, S. A. Vorobyov, and C. Tellambura, "Power allocation in wireless multi-user relay networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 5, pp. 2535–2545, May 2009.

[26] A. Ikhlef and R. Schober, "Joint source-relay optimization for fixed receivers in multi-antenna multi-relay networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 1, pp. 62–74, Jan. 2014.

[27] K. Gomadam, V. R. Cadambe, and S. A. Jafar, "A distributed numerical approach to interference alignment and applications to wireless interference networks," *IEEE Trans. Inf. Theory*, vol. 57, no. 6, pp. 3309–3322, Jun. 2011.

[28] S. Kumar and K. Rajawat, "Distributed interference alignment for MIMO cellular network via consensus ADMM," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Dec. 2016, pp. 460–464.

[29] E. Visotsky and U. Madhow, "Optimum beamforming using transmit antenna arrays," in *Proc. IEEE 49th Veh. Technol. Conf.*, May 1999, pp. 851–856.

[30] M. Fukushima, "Application of the alternating direction method of multipliers to separable convex programming problems," *Comput. Optim. Appl.*, vol. 1, no. 1, pp. 93–111, Oct. 1992.

[31] S. Boyd, L. Xiao, A. Mutapcic, and J. Mattingley. (2008). *Notes on Decomposition Methods*. [Online]. Available: https://stanford.edu/class/ee364b/lectures/decomposition_notes.pdf

[32] M. Grant and S. Boyd. (Mar. 2014). *CVX: MATLAB Software for Disciplined Convex Programming, Version 2.1*. [Online]. Available: http://cvxr.com/cvx

[33] V. G. Douros and G. C. Polyzos, "Review of some fundamental approaches for power control in wireless networks," *Comput. Commun.*, vol. 34, no. 13, pp. 1580–1592, Aug. 2011.

[34] J. Mattingley and S. Boyd, "Real-time convex optimization in signal processing," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 50–61, May 2010.

[35] A. Afana, V. Asghari, A. Ghrayeb, and S. Affes, "On the performance of cooperative relaying spectrum-sharing systems with collaborative distributed beamforming," *IEEE Trans. Commun.*, vol. 62, no. 3, pp. 857–871, Mar. 2014.

[36] S. Guruacharya, H. Tabassum, and E. Hossain, "SINR outage evaluation: Saddle point approximation using normal inverse Gaussian distribution," *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 591–605, Jan. 2018.

[37] E. Song, Q. Shi, M. Sanjabi, R. Sun, and Z.-Q. Luo, "Robust SINR-constrained MISO downlink beamforming: When is semidefinite programming relaxation tight?" in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 3096–3099.

[38] T.-H. Chang, W.-K. Ma, and C.-Y. Chi, "Worst-case robust multiuser transmit beamforming using semidefinite relaxation: Duality and implications," in *Proc. Conf. Rec. 45th Asilomar Conf. Signals, Syst. Comput. (ASILOMAR)*, Nov. 2011, pp. 1579–1583.

CENK M. YETIS received the B.Sc. degree in electronics engineering from Isik University, Istanbul, Turkey, in 2001, the M.Sc. degree in telecommunications engineering from Istanbul Technical University (ITU), in 2004, and the Ph.D. degree in satellite communications and remote sensing from ITU in 2010. From 2003 to 2007, he was with Avea, one of the top three wireless services providers in Turkey, where he held rotational responsibilities in operation and planning groups. From 2007 to 2010, he was a Visiting Researcher at The Ohio State University and The University of California at Irvine. From 2010 to 2016, he held academic positions in universities in Hong Kong, Singapore, and Turkey. In 2016, he joined the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan as a Post-Doctoral Fellow to extend his research and teaching activities. His research interests include signal processing, information theory, communication theory, and optimization theory for wireless communications. He received The Scientific and Technological Research Council of Turkey scholarship from 2005 to 2009. In 2017, he received a two-year Academia Sinica fellowship.

RONALD Y. CHANG (M'12) received the B.S. degree in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2000, the M.S. degree in electronics engineering from National Chiao Tung University, Hsinchu, in 2002, and the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 2008. From 2002 to 2003, he was with the Industrial Technology Research Institute, Hsinchu. In 2008, he was a Research Intern at Mitsubishi Electric Research Laboratories, Cambridge, MA, USA. In 2009, he was involved with the NASA Small Business Innovation Research projects. Since 2010, he has been with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan, where he is currently an Associate Research Fellow (Associate Professor). He was a Visiting Scholar with the Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA, in 2018. His research interests include wireless communications and networking. He was a recipient of the Best Paper Award from the IEEE Wireless Communications and Networking Conference in 2012 and the Outstanding Young Scholar Award from the Ministry of Science and Technology, Taiwan, in 2015 and 2017, respectively. He was an Exemplary Reviewer of the IEEE COMMUNICATIONS LETTERS in 2012, the IEEE TRANSACTIONS ON COMMUNICATIONS in 2015, and the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS in 2017.

• • •