Joint Transceiver Design for Noisy-Sensing Decision-Fusion Networks through Block Coordinate Descent Optimization

Yang Liu, Xuanxuan Lu, Jing Li (Tiffany)

Electrical and Computer Engineering Department, Lehigh University, Bethlehem, PA 18015
Email: {yal210, xul211, jingli}@ece.lehigh.edu

Abstract—This paper considers joint transceiver design for a wireless sensor network where multiple sensors, each equipped with $N$ antennas, transmit individually-contaminated data to a fusion center that is equipped with $M$ antennas. Rather than assuming an overall power provision among all the sensors, we consider a more realistic condition of individual power constraint. Under the mean square error (MSE) criterion, the joint beamforming design problem can be formulated as a nonconvex optimization problem, one that is not directly solvable using conventional tools. To attack it, we first propose a 2-block coordinate descent (2-BCD) method that iteratively designs all the precoders (at all the sensors) and the joint postcoder (at the fusion center). We identify the convexity of the subproblems, develop effective ways to solve both subproblems, and examine the overall convergence. Based on the insights gained in the 2-BCD method, we further propose two new algorithms, layered-BCD and cyclic-BCD, both of which turn out to provide better performance, considerably lower complexity and faster convergence than 2-BCD. Extensive numerical results are presented to verify our analytical findings.

Index Terms—wireless sensor network, MMSE, beamforming, transceiver, block coordinate descent method.

I. INTRODUCTION

Notations: The following notations will be used in the manuscript.

- bold lowercase: complex vectors
- bold uppercase: denote complex matrices
- $O_{M \times N}$: the all-zero matrix of dimension $M \times N$
- $I_M$: the identity matrix of dimension $M \times M$
- $A^T$: transpose of $A$
- $A^*$: conjugate of $A$
- $A^H$: Hermitian transpose of $A$
- $\mathbb{A}$: Moore-Penrose pseudoinverse of $A$
- $\text{Tr}{}$: trace operation of a matrix
- $| \cdot |$: modulus of a complex scalar
- $\| \cdot \|_2$: $l_2$-norm of a complex vector
- $\text{vec}{}$: vectorization operation of a matrix
- $A \succ 0$: $A$ is Hermitian and positive definite
- $A \succeq 0$: $A$ is Hermitian and positive semi-definite
- $\text{diag}()$: form block diagonal matrix by arguments

Consider a typical wireless sensor network (WSN) comprised of a fusion center (FC) and a large number of sensors that are spatially distributed and wirelessly connected to provide surveillance to a region of interest. Depending on the WSN density, it is possible for multiple sensors in the same neighborhood to be capable of observing and detecting the same physical event. After harvesting information from the environment, these sensors will transmit their (possibly distorted) observations to the fusion center (FC) to perform data processing and decision fusion [4]-[8]. A central problem underlying these systems is how to design good transceivers such that the multiple sensors and the fusion center can collaboratively accomplish the sensing, communication, and decision fusion task in a most efficient and trust-worthy manner.

When the sensors and the fusion center are all equipped with multiple antennas, this problem may be regarded as one of cooperative multi-input multi-output (MIMO) beamforming design, and researchers have been tackling it from various perspectives. For example, several studies [4]-[6] targeted reducing the dimensionality (i.e. rate $< 1$) as a design criterion. They proposed ways to achieve compressed beamforming, and considered the cases when the channels between the sensors and the fusion center are assumed perfect with no fading nor noise uncertainty [4], [5] and when the transmission channels are non-perfect [6]. The majority of the studies in the literature, however, focus on classic beamforming (i.e. rate 1). The goal is to achieve the best communication reliability and efficiency [7], [8], [9]-[13], which is typically assessed in terms of mean square error (MSE, the most common criterion), the end-to-end mutual information, or the equivalent signal-to-noise ratio (SNR). From a different perspective, the MIMO multi-sensor decision-fusion system may also be regarded as some form of multiple relay system [9], [11], and a large number of exciting papers exist in the literature, see, for example, [10], [12], [13] and the references therein.

This paper considers transceiver design for a rather general MIMO sensing model that appears more difficult than the models studied in many previous works. Our model allows all the sensor observations to be individually corrupted, all the communication channels to be individually faded, and each sensor-FC transmission to have its own power constraint rather than an overall constraint. (The combination of the first and the last assumption appears to have particularly complicated the design issue.)

Two recent studies are particularly relevant to our work. The
first, by Xiao, Cui, Luo and Goldsmith, was the first to present this very general system model [7]. It classified a variety of interesting cases subsumed in the general model, and presented beamforming solutions to several special cases such as the noiseless sensor-FC channel case and the no-intersymbol-interference (no-ISI) channel case. The second [8], by Bebbahani, Eltawil and Jafarkhani, developed a very useful type of iterative BCD beamforming optimization method that is applicable to the general model. The requirement thereof is that, when solving the Karush-Kuhn-Tucker (KKT) condition, certain matrices (\(M_i\)) must have a full rank and can therefore take a direct matrix inverse [3] [8].

This paper considers the same general model discussed in [7], [8]. The goal is to develop effective transceivers by jointly optimizing the precoders at the sensors and the postcoder at the fusion center. It is shown that the overall problem is a nonconvex optimization problem that cannot be readily handled by conventional tools. Making use of the block coordinate descent (BCD) framework, we successfully develop three effective methods for transceiver design. Our specific contributions include:

1) Using MSE as a design criterion, we first propose a 2-block coordinate descent (2-BCD) method that decomposes the original problem into two subproblems. We show that the subproblem of optimizing the postcoder, given the precoders, is a minimum mean square error (MMSE) problem whose closed-form solution turns out to be the Wiener filter. We show that the other subproblem, optimizing the precoders given the postcoder, is convex, and can be reformulated as a convex quadratic constraint quadratic problem (QCQP). It is shown in [7] that in the special case where the sensor-FC channels do not have intersymbol-interference (ISI) (i.e. the sensor-FC channel matrix is an identity or diagonal matrix), this second subproblem can be reformulated as semidefinite programming (SDP) problem. Here we develop a stronger conclusion, showing that the SDP problem holds even with arbitrary sensor-FC channel matrices. We further show that this particular convex QCQP problem can be transformed into a second order cone problem (SOCP), a special SDP that promises more efficient numerical methods than general SDP. We provide rigorous convergence analysis for this 2-BCD algorithm, not only showing that the optimization target is converging, but also showing that the optimization result has limit points, every one of which converges to some stationary point.

2) In addition to numerical methods, we also attack the second subproblem by approximating it into \(L\) subsubproblems of precoder design (where \(L\) is the number of sensors), and developing (nearly fully) analytical solution for each precoder via the KKT conditions. (This helps pave the way for the proposition of two more effective algorithms for optimizing precoders.) It should be noted that, although the technique of checking the KKT condition for each separate beamformer is rather standard and has also been adopted in several previous papers (e.g. [12], [13] and [6]), we are able to carry out the computation to the very end and fully solved the problem by clearly describing the solution structure and deriving the exact closed-form solutions. Specifically, we explicitly obtain the equivalent conditions for judging the positiveness of the Lagrange multipliers, and, in the case of zero-Lagrange-multipliers, we derive the equivalent conditions for identifying the optimality of the solution via pseudoinverse. These exact results, and especially the case of the zero-Lagrange-multiplier, are not discussed previously in the literature.

3) Based on the insights gained in 2-BCD, we propose a layered-BCD algorithm, where the original problem of jointly designing all the precoders and the postcoder is transformed into two layers of embedded BCD structure: the outer layer takes the form of 2-BCD, and inner layer handles the the subproblem of designing all the precoders using an \(L\)-BCD algorithm (where \(L\) is the number of sensors). We prove that with the inner-layer \(L\)-BCD algorithm sufficiently converging (after an infinitely large number of iterations), the entire layered-BCD algorithm can also guarantee a successful convergence to some stationary point. For practical applications, we also analyze the performance of the layered-BCD algorithm under the limitation of a finite number of inner iterations, and arrive at a convergence result that is only slightly weaker than that of the infinite inner iterations. It is interesting to note that the method presented in [8] turns out to be a special case of our layered-BCD algorithm with the inner iteration number set to 1, but our results are more comprehensive and include concrete solutions to the case when matrix \(M_i\) is rank-deficient and not invertible (definition of \(M_i\) will be introduced later).

4) We also propose a cyclic-BCD algorithm, which decomposes the original optimization problem into \((L+1)\) blocks and updates the postcoder and the \(L\) precoders in a particular cyclic order of period 2\(L\). We present nearly closed-form solutions to this approach, evaluate its convergence property, and further provide an approximation to it. This cyclic-BCD algorithm not only appears to have the lowest complexity, but its approximated version also turns out to exhibit a surprisingly fast convergence and a superb performance that all the other algorithms are incapable of!

The rest of the paper is organized as follows: Section II introduces the system model of the MIMO noisy-sensing network with a fusion center and formulates the overall joint beamforming design problem. Section III discusses the proposed 2-BCD beamforming design approach, and analyzes the convexity and the convergence properties. Section IV discusses the further decomposition of the precoder-design problem, and proposes a layered-BCD method and a cyclic-BCD method, both of which outperform the 2-BCD approach in terms of complexity and convergence. Section V provides simulation verification and Section VI concludes the article.
II. SYSTEM MODEL

Consider a centralized wireless sensor network with $L$ sensors and one fusion center where all the nodes are equipped with multiple antennae, as shown in Figure 1. Let $N_i$ ($i = 1, 2, \cdots, L$) be the number of antenna provisioned to the $i$-th sensor, and let $M$ be the number of antennas provisioned to the fusion center. Let $s$ be the common source vector observed by all the sensors. Due to interference from the surrounding environment or thermal noise from the sensor device, the observed signals at the sensor are typically contaminated. To be general, the source $s$ is assumed to be a complex source vector of dimension $K$, i.e. $s \in \mathbb{C}^{K \times 1}$, and the $i$-th sensor may observe some form of environment or thermal noise from the sensor device, the channel fading matrices $\mathbf{H}_i \in \mathbb{C}^{M \times N_i}$, where the subscript $i$ denotes the sensor index. Since all the sensors are spatially distributed, it is reasonable to assume that the channel fading matrices $\mathbf{H}_i$ are mutually independent. The channel noise is modeled as an additive white circular Gaussian vector $\mathbf{n}_i \in \mathbb{C}^{M \times 1}$ with $\mathbf{n}_i \sim \mathcal{CN}(0, \Sigma_i)$. The fusion center, after collecting all the results, applies a linear postcoder, $\mathbf{G}^H \in \mathbb{C}^{K \times M}$, to retrieve the original source $s$.

Mathematically, the signal transmitted by the $i$-th sensor takes the form of $\mathbf{F}_i (\mathbf{K}_i s + \mathbf{n}_i)$. The output $\hat{s}$ of the postcoder at the fusion center is given by:

$$
\hat{s} = \mathbf{G}^H \mathbf{r} = \mathbf{G}^H \left( \sum_{i=1}^{L} \mathbf{H}_i \mathbf{F}_i (\mathbf{K}_i s + \mathbf{n}_i) + \mathbf{n}_0 \right)
$$

where

$$
\mathbf{G}^H \left( \sum_{i=1}^{L} \mathbf{H}_i \mathbf{F}_i \mathbf{K}_i \right) s + \mathbf{G}^H \left( \sum_{i=1}^{L} \mathbf{H}_i \mathbf{F}_i \mathbf{n}_i + \mathbf{n}_0 \right). \tag{2}
$$

Comment II.1. It should be pointed out that the whiteness assumption of the Gaussian noise $\mathbf{n}_0$ at the receiver does not undermine any generality. Indeed if $\mathbf{n}_0 \sim \mathcal{CN}(0, \Sigma_0)$ is colored, i.e. the covariance matrix $\Sigma_0$ is not a diagonal matrix, we can redefine $\hat{\mathbf{r}} \triangleq \Sigma_0^{-\frac{1}{2}} \mathbf{r}$, $\mathbf{H}_i \triangleq \Sigma_0^{-\frac{1}{2}} \mathbf{H}_i$ and $\tilde{\mathbf{n}}_0 \triangleq \Sigma_0^{-\frac{1}{2}} \mathbf{n}_0$; then the received signal can be written equivalently as

$$
\hat{\mathbf{r}} = \left( \sum_{i=1}^{L} \mathbf{H}_i \mathbf{F}_i \left( \mathbf{K}_i s + \mathbf{n}_i \right) \right) + \tilde{\mathbf{n}}_0, \tag{4}
$$

where $\tilde{\mathbf{n}}_0 \sim \mathcal{CN}(0, \mathbf{I}_M)$ is a white noise. This goes back to the model in (2).

In this paper, we take the mean square error as a figure of merit to guide the design of the precoders $\mathbf{F}_i$ and the postcoder $\mathbf{G}$. The mean square error matrix $\Phi$ is defined as

$$
\Phi = \mathbb{E} \{ (\mathbf{s} - \hat{\mathbf{s}}) (\mathbf{s} - \hat{\mathbf{s}})^H \}. \tag{5}
$$

By assuming that the source signal $\mathbf{s}$ has a zero mean and a covariance matrix $\Sigma_s = \mathbb{E} \{ \mathbf{s} \mathbf{s}^H \}$, and by plugging (2) into (5), we can express the MSE matrix $\Phi$ as a function of $\{\mathbf{F}_i\}$ and $\mathbf{G}$ as:

$$
\Phi \left( \{\mathbf{F}_i\}_{i=1}^{L}, \mathbf{G} \right) = \mathbf{G}^H \left( \sum_{i=1}^{L} \mathbf{H}_i \mathbf{F}_i \mathbf{K}_i \right) \Sigma_s \left( \sum_{i=1}^{L} \mathbf{H}_i \mathbf{F}_i \mathbf{K}_i \right)^H \mathbf{G}
$$

$$
- \mathbf{G}^H \left( \sum_{i=1}^{L} \mathbf{H}_i \mathbf{F}_i \mathbf{K}_i \right) \Sigma_s \Sigma_s \left( \sum_{i=1}^{L} \mathbf{H}_i \mathbf{F}_i \mathbf{K}_i \right)^H \mathbf{G}
$$

$$
+ \sum_{i=1}^{L} \mathbf{G}^H \mathbf{H}_i \mathbf{F}_i \Sigma_s \mathbf{F}_i^H \mathbf{H}_i^H \mathbf{G} + \sigma_0^2 \mathbf{G}^H \mathbf{G} + \Sigma_s. \tag{6}
$$

The total MSE is then given by

$$
\text{MSE} \left( \{\mathbf{F}_i\}_{i=1}^{L}, \mathbf{G} \right) \triangleq \text{Tr} \left\{ \Phi \left( \{\mathbf{F}_i\}_{i=1}^{L}, \mathbf{G} \right) \right\}. \tag{7}
$$

We consider the realistic case where each sensor has its own transmission power constraint. This leads to the following power constraint: $\mathbb{E} \{ \| \mathbf{F}_i (\mathbf{K}_i s + \mathbf{n}_i) \|^2 \} = \text{Tr} \{ \mathbf{F}_i (\mathbf{K}_i \Sigma_s \mathbf{F}_i^H + \Sigma_i) \mathbf{F}_i^H \} \leq P_i$, $i \in \{1, \cdots, L\}$. The overall beamformer design problem can then be formulated as the following optimization problem:

$$
\text{min} \quad \text{MSE} \left( \{\mathbf{F}_i\}_{i=1}^{L}, \mathbf{G} \right), \tag{8a}
$$

s.t. $\text{Tr} \{ \mathbf{F}_i (\mathbf{K}_i \Sigma_s \mathbf{F}_i^H + \Sigma_i) \mathbf{F}_i^H \} \leq P_i$, $i \in \{1, \cdots, L\}. \tag{8b}$

Fig. 1: Multi-Sensor System Model
The above problem is nonconvex, which can be easily seen by checking the special case where \( \{F_i\}_{i=1}^L \) and \( G \) are all scalars. Such a problem tends to have numerous local minimaums and the closed-form solution of the global minimum is usually inaccessible.

### III. Two-Block Coordinate Descent (2-BCD)

The intractability of (P0) motivates us to consider the framework of block coordinate descent (BCD) method\(^{(15)}\)\(^{(16)}\). Also known as the alternative minimization algorithm (AMA)\(^{(14)}\) or the Gauss-Seidel(GS) algorithm\(^{(15)}\)\(^{(18)}\). BCD provides an approximated means to solve difficult non-convex problems: when the original problem is intractable with all the variables jointly considered, one may partition the variables into separate groups, and optimize each group (with the others being fixed) alternatively. In general, appropriate decomposition can lead to efficiently solvable subproblems and may also provide opportunities for parallel computation.

To put (P0) in perspective, we first study a 2-BCD method that decouples the design of the postcoder \( G \) (conditioned on the precoders), thereafter referred to as (P1), from the design of all the precoders \( \{F_i\}_{i=1}^L \) (conditioned on the postcoder), thereafter referred to as (P2).

#### A. P1: Optimizing \( G \) given \( \{F_i\} \)

With \( \{F_i\}_{i=1}^L \) being given, optimizing MSE with respect to \( G \) becomes a non-constrained quadratic optimization problem (P1):

\[
\text{(P1)}: \min_G \text{Tr}\left\{ \Phi\left(G; \{F_i\}_{i=1}^L\right) \right\}.
\]

(P1) is in fact an MMSE receiver problem. By equating the derivative \( \frac{\partial}{\partial G}\text{MSE}(G) \) with zero, the closed-form solution \( G^*_{(p1)} \) to (9) can be readily obtained as the well-known Wiener filter\(^{(21)}\):

\[
G^*_{(p1)} = \left( \sum_{i=1}^L H_i F_i K_i \right) \Sigma_n \left( \sum_{i=1}^L H_i F_i K_i \right)^H + \Sigma_n \right]^{-1} \sum_{i=1}^L H_i F_i K_i \Sigma_n,
\]

where \( \Sigma_n \) is given in (3).

#### B. P2: Optimizing \( \{F_i\} \) given \( G \)

With \( G \) being fixed, the subproblem (P2) minimizes MSE with respect to \( \{F_i\}_{i=1}^L \):

\[
\text{(P2)}: \min_{\{F_i\}_{i=1}^L} \text{Tr}\left\{ \Phi\left( \{F_i\}_{i=1}^L; G \right) \right\},
\]

\[
\text{s.t. } \text{Tr}\left\{ F_i (K_i \Sigma_n K_i^H + \Sigma_n) F_i^H \right\} \leq P_i, \quad i \in \{1, \cdots, L\}.
\]

Below we discuss the convexity of (P2).

**Theorem 1.** (P2) is convex with respect to \( \{F_i\}_{i=1}^L \).

**Proof:** First consider the function \( f(X) : C^{m \times n} \rightarrow \mathbb{R} \)

\[
f(X) = \text{Tr}\{A^H X \Sigma X H^H A\}
\]  

(12)

where the constant matrices \( A \) and \( \Sigma \) have appropriate dimensions and \( \Sigma \) is Hermitian and positive semi-definite.

By the identities \( \text{Tr}\{AB\} = \text{Tr}\{BA\} \) and \( \text{Tr}\{ABCD\} = \text{vec}^T(D^T)[C^T \otimes A] \text{vec}(B) \), \( f(X) \) can be equivalently written as

\[
f(X) = \text{Tr}\{AA^H X \Sigma X H^H A\}
\]

\[
= vec^H(X) \left[ \Sigma^* \otimes (AA^H) \right] vec(X). \quad (13)
\]

Clearly, \( AA^H \) is positive semi-definite, and so is \( \Sigma^* \) by assumption. Thus their Kronecker product should also be positive semi-definite. Indeed given two Hermitian matrices \( A_{m \times m} \) and \( B_{n \times n} \) having eigenvalues \( \{\lambda_i(A)\}_{i=1}^m \) and \( \{\lambda_j(B)\}_{j=1}^n \), respectively, the eigenvalues of their Kronecker product \( A \otimes B \) are given by \( \{\lambda_i(A)\lambda_j(B)\}_{i=1}^m{n}_{j=1}^n \)\(^{(20)}\). Consequently, when \( A \) and \( B \) are both positive semi-definite, all the eigenvalues of \( A \otimes B \) are also non-negative. It then follows that (13) is actually a convex homogeneous quadratic function of \( \text{vec}(X) \), which suggests that \( f(X) \) is convex.

Now replace the original variable \( X \) by \( \sum_{i=1}^L (H_i F_i K_i) \) with \( \{H_i\} \) and \( \{K_i\} \) being constant, and recall the fact that affine operation preserves convexity\(^{(22)}\), the quadratic term

\[
\text{Tr}\left\{ G^H \left( \sum_{i=1}^L H_i F_i K_i \right) \Sigma_n \left( \sum_{i=1}^L H_i F_i K_i \right)^H G \right\}
\]

in the objective function (P2) is therefore convex with respect to \( \{F_i\}_{i=1}^L \).

By the same reasoning, \( \text{Tr}\left\{ \sum_{i=1}^L G^H H_i F_i \Sigma_n H_i^H F_i^H G \right\} \) in the objective and \( \text{Tr}\left\{ F_i (K_i \Sigma_n K_i^H + \Sigma_n) F_i^H \right\} \) in the constraints, are also convex with respect to \( \{F_i\}_{i=1}^L \).

The remaining terms in the objective (17a) (see (5)) are affine functions of \( \{F_i\}_{i=1}^L \), and therefore convex.

Hence, the optimization problem (P2) has a convex objective and a set of convex constraints, which means (P2) is convex with respect to \( \{F_i\}_{i=1}^L \).

After recognizing its convexity, we now reformulate the subproblem (P2) into a standard convex quadratic constraint quadratic programming (QCQP) form, which leads to a second order cone programming(SOCP) presentation and helps simplify our subsequent exposition. To this end, we introduce the following notations:

\[
f_i \triangleq \text{vec}(F_i);
\]

\[
g \triangleq \text{vec}(G);
\]

\[
A_{ij} \triangleq (K_i \Sigma_n K_i^H)^T \otimes (H_i^H \Sigma_n \Sigma_n H_i^H);
\]

\[
B_i \triangleq (K_i \Sigma_n)^T \otimes H_i;
\]

\[
C_i \triangleq \Sigma_i^* \otimes (H_i^H \Sigma_n \Sigma_n H_i^H).
\]

By the notations introduced above and repeatedly utilizing the identity \( \text{Tr}\{ABCD\} = \text{vec}^T(D^T)[C^T \otimes A] \text{vec}(B) \), \( f(X) \) we can rewrite the MSE in (P2) as

\[
\text{MSE}\left( \{f_i\}_{i=1}^L \mid g \right) = \sum_{i=1}^L \sum_{j=1}^L f_i^H A_{ij} f_j - 2 \text{Re}\left( \sum_{i=1}^L g_i^H B_i f_i \right)
\]

\[
+ \sum_{i=1}^L f_i^H C_i f_i + \sigma_0^2\|g\|^2 + \text{Tr}\{\Sigma_n\}. \quad (15)
\]

To further simplify the problem formulation, we introduce the following definitions:
Algorithm 1: 2-BCD Algorithm to Solve (P0)

Input: $\Sigma_0$, $\{\Sigma_i\}_{i=1}^L$, $\{H_i\}_{i=1}^L$, $\{K_i\}_{i=1}^L$, $\{P_i\}_{i=1}^L$, $\sigma_0^2$, $\epsilon_0$ (or $M_0$);
Output: $\{F_i\}_{i=1}^L$ and $G$.

1 Initialization: Randomly generate feasible $\{F_i^{(0)}\}_{i=1}^L$, $i \in \{1, \cdots, L\}$; Compute $G^{(0)}$ using (10); $\text{MSE}^{(0)} = \text{MSE}((F_i^{(0)})_{i=1}^L, G^{(0)}); \Delta = \infty$;

2 for $j = 1; (\Delta > \epsilon_0)$ (or $j < M_0$); $j++$ do

3 \hspace{1cm} With $G^{(j-1)}$ fixed, solve (P2’) and obtain $\{F_i^{(j)}\}_{i=1}^L$;

4 \hspace{1cm} With $\{F_i^{(j)}\}_{i=1}^L$ fixed, compute $G^{(j)}$ using (10).

5 \hspace{1cm} Calculate $\text{MSE}^{(j)} = \text{MSE}((F_i^{(j)})_{i=1}^L, G^{(j)}))$ by (7);

6 \hspace{1cm} calculate $\Delta = \text{MSE}^{(j-1)} - \text{MSE}^{(j)}$;

7 end

8 Return $\{F_i^{(L)}\}_{i=1}^L$ and $G^{(L)}$.

subsequence will converge. Consider the optimization problem

$$\min \{ f(x) | x \in X \}$$

with $f(\cdot)$ being continuously differentiable and the domain $X$ being closed and nonempty. A point $x_0 \in X$ is a stationary point if and only if $\nabla f(x_0)(x - x_0) \geq 0, \forall x \in X$, where $\nabla f(x_0)$ denotes the gradient of $f$ at $x_0$.

The majority of the beamforming papers using the BCD method have only pointed out that the objective function to be optimized is converging (i.e. the target MSE is monotonically decreasing along each BCD iteration, and lower bounded by 0). Here we deepen the convergence analysis by demonstrating that not only does the objective function MSE converge, but there exists a subsequence of the achieved optimization result, $\{\{F_i\}_{i=1}^L, G\}$, which is also converging (to some stationary point). Our convergence analysis makes use of the following important conclusion from [18].

Lemma 1 (Corollary 2 in [18]). Consider the following optimization problem

$$\min f(x) \quad (19a)$$

$$\text{s.t. } x \in X = X_1 \times X_2 \times \cdots \times X_m, \quad (19b)$$

where $f(\cdot)$ is continuously differentiable, and $X$ is the Cartesian product of closed, nonempty and convex subsets $X_i$, for $i = 1, \cdots, m$. Suppose that the solution sequence $\{x^{(k)}\}$ to problem (19) generated by a two-block coordinate descent method (m = 2) has limit points. Then every limit point of $\{x^{(k)}\}$ is a stationary point of the problem (19).

Theorem 2. The objective sequence $\{\text{MSE}^{(j)}\}_{j=0}^{\infty}$ generated by Algorithm 1 is monotonically decreasing. If $K_i \Sigma_i K_i^H \succ 0$ or $\Sigma_i \succ 0$ for all $i \in \{1, \cdots, L\}$, by setting $\epsilon_0 \to 0$ (or $N_0 \to \infty$), the solution sequence $\{\{F_i^{(j)}\}_{i=1}^L, G^{(j)}\}_{j=1}^{\infty}$ generated by Algorithm 1 has limit points and every limit point of $\{\{F_i^{(j)}\}_{i=1}^L, G^{(j)}\}_{j=1}^{\infty}$ is a stationary point of (P0).

Proof: That MSE keeps decreasing is rather obvious, as each block update solves a minimization problem.

Clearly the objective function of (P0) is continuously differentiable. Let $X_1 = \mathbb{C}^{M \times K}$ and $X_{(i+1)} = \{X \in$
\( \mathbb{C}^{N \times J} \{ \mathbf{X}(\mathbf{K}, \mathbf{\Sigma}_0, \mathbf{K}_t^H + \mathbf{\Sigma}_t) \mathbf{X}^H \} \leq \mathbf{P}_i \}, \) for \( i = 1, \ldots, L. \) We see that the condition \((19)\) on \( X \) of Lemma \((1)\) is satisfied. 

Under the strict positive definiteness assumption of \( \mathbf{K}, \mathbf{\Sigma}_0, \mathbf{K}_t^H \) or \( \mathbf{\Sigma}_0, \) we have \((\mathbf{K}, \mathbf{\Sigma}_0, \mathbf{K}_t^H + \mathbf{\Sigma}_i) \succ 0 \) and thus \((\mathbf{K}, \mathbf{\Sigma}_0, \mathbf{K}_t^H + \mathbf{\Sigma}_i)^T \otimes \mathbf{I}_{N_i} \succ 0 \) for all \( i \in \{1, \ldots, L\}. \) This means that the null space of \((\mathbf{K}, \mathbf{\Sigma}_0, \mathbf{K}_t^H + \mathbf{\Sigma}_i)^T \otimes \mathbf{I}_{N_i} \) is \( \{0\} \) and consequently \( f_i \) has to be bounded to satisfy power constraint. Equivalently, this means that \( \lambda_i \) is bounded for all \( i \in \{2, \ldots, L + 1\}. \) Since the feasible set for each \( f_i \) is bounded, by Bolzano-Weierstrass theorem, there exists a convergent subsequence \((\{f_{i,j}\})\) \( \{f_{j,k}\} \) \( \{f_{i,k}\} \) \( \{f_{i,k}\} \). Since algorithm II is a two block coordinate descent procedure, using the result from Lemma I we conclude that any limit point of \((\{f_{i,j}\}) \) is a stationary point of (P0).

IV. MULTI-BLOCK COORDINATE DESCENT

In the proposed 2-BCD algorithm, although we have identified the convexity of subproblem (P2) and transformed it into a standard SOCP problem (P2'), its closed-form solution is still inaccessible. The complexity for solving (P2) or (P2') is \( \mathcal{O}\left(\left(\sum_{i=1}^{L} N_i J_i\right)^3\right) \), where \( \left(\sum_{i=1}^{L} N_i J_i\right) \) is the length of the input variables. This implies that when the sensor network under consideration has a large number of sensors and/or antenna, the complexity for solving (P2) can be rather daunting. This motivates us to search for more efficient ways to solve (P2).

A. Further Decoupling of P2 and Closed-Form Solution

The ideal way to solve a problem is to obtain its optimal solution in a closed form. To gain insight into the solutions of problem (P2), we examine its Karush-Kuhn-Tucker conditions [22].

We first compute the Lagrangian of (P2):

\[
\mathcal{L}(P2)'(\{f_i\}_{i=1}^{L}, \{\mu_i\}_{i=1}^{L}) = f^H (\mathbf{A} + \mathbf{C}) f - 2 \text{Re} \{\mathbf{g}^H \mathbf{B} f\} + c
\]

+ \sum_{i=1}^{L} \mu_i (f^H \mathbf{D}_i f - P_i),
\]

where the variables \( \mu_i \) are the Lagrangian multipliers. The KKT conditions for (P2) are given by:

\[
\sum_{j=1}^{L} \mathbf{A}_{ij} f_j - \mathbf{B}_{ij}^H \mathbf{g} + \mathbf{C}_i f_i + \mu_i f_i = 0; \quad i \in \{1, \ldots, L\} \tag{21a}
\]

\[
f^H f_i \leq P_i; \quad i \in \{1, \ldots, L\} \tag{21b}
\]

\[
\mu_i(f^H f_i - P_i) = 0; \quad i \in \{1, \ldots, L\} \tag{21c}
\]

\[
\mu_i \geq 0; \quad i \in \{1, \ldots, L\} \tag{21d}
\]

Since the first-order condition \((21a)\) involves \( f_j \) for all \( j \in \{1, \ldots, L\}, \) \( f_i \) can be determined only if the knowledge of \( \{f_j\}_{j \neq i} \) is available. However, the tangling \( f_i \)’s make the KKT conditions unsolvable. This observation motivates the idea of further decomposition of the subproblem (P2) using the BCD idea. Naturally, for a given \( \mathbf{G}, \) instead of optimizing all the \( f_i \)’s in a single batch, we can simplify the task by optimizing one specific \( f_i \) at a time (with the others being fixed) and iteratively going through all of them. By introducing the notation \( q_i \triangleq \sum_{j=1}^{L} A_{ij} f_j, \) each subproblem (P2i’) of (P2) can be written as

\[
(P2i') : \min_{f_i} f_i^H (\mathbf{A}_{ii} + \mathbf{C}_i) f_i + 2 \text{Re} \{q_i^H f_i\} - 2 \text{Re} \{\mathbf{g}^H \mathbf{B}_i f_i\} \tag{22a}
\]

s.t. \( f_i^H f_i \leq P_i. \tag{22b} \]

Our problem now boils down to solving the simpler problem (P2i’), for \( i = 1, \ldots, L. \) The following theorem provides an almost closed-form solution to (P2i’). The only reason that this is not a fully closed-form solution is because it may involve numerically determining the value of a positive real number.

Theorem 3. We assume \( \mathbf{K}, \mathbf{\Sigma}_0, \mathbf{K}_t^H \succ 0 \) or \( \mathbf{\Sigma}_i \succ 0, \) \( i = 1, \ldots, L. \) The optimal solution \( f_i^* \) to Problem (P2i’) is given by either of the following two equations:

\[
f_i^* = (\mathbf{A}_{ii} + \mathbf{C}_i + \mu_i^* \mathbf{E}_i)^{-1} (\mathbf{B}_{ii}^H \mathbf{g} - q_i), \tag{23a}
\]

or \( f_i^* = \mathbf{E}_i^{-\frac{1}{2}} (\mathbf{A}_{ii} + \mathbf{C}_i) \mathbf{E}_i^{-\frac{1}{2}} \mathbf{E}_i^{-\frac{1}{2}} (\mathbf{B}_{ii}^H \mathbf{g} - q_i), \tag{23b}\)

where \( \mu_i^* \) is a positive value lying in the interval \([lbd_i, ubd_i]\). (1bdi and ubdi will be explicitly given in Lemma 2, which comes later) and can be numerically determined.

Proof: From the assumption, we get \((\mathbf{K}, \mathbf{\Sigma}_0, \mathbf{K}_t^H + \mathbf{\Sigma}_i) \succ 0. \) Thus \( \mathbf{E}_i = (\mathbf{K}, \mathbf{\Sigma}_0, \mathbf{K}_t^H + \mathbf{\Sigma}_i) \otimes \mathbf{I}_{N_i} \succ 0. \) We introduce the following notations

\[
\tilde{f} \triangleq \mathbf{E}_i^{\frac{1}{2}} f_i; \tag{24a}
\]

\[
\mathbf{M}_i \triangleq \mathbf{E}_i^{-\frac{1}{2}} (\mathbf{A}_{ii} + \mathbf{C}_i) \mathbf{E}_i^{-\frac{1}{2}} = \mathbf{U}_i \mathbf{A}_i \mathbf{U}_i^H; \tag{24b}
\]

\[
\mathbf{b}_i \triangleq \mathbf{E}_i^{-\frac{1}{2}} (\mathbf{B}_{ii}^H \mathbf{g} - q_i); \tag{24c}
\]

\[
\mathbf{p}_i \triangleq \mathbf{U}_i^H \mathbf{b}_i; \tag{24d}
\]

where the columns \( \mathbf{u}_{i,j} \) of \( \mathbf{U}_i \) are the eigenvector associated to the eigenvalue \( \lambda_{i,j} \triangleq [\lambda_i]_{i,j}. \) Without loss of generality, we assume that the eigenvalues of \( \mathbf{M}_i \) are arranged in a decreasing order and that \( \mathbf{M}_i \) has rank \( r_i, \) where \( r_i \leq J_i N_i. \) That is, \( \lambda_{i,1} \geq \ldots \geq \lambda_{i,r_i} > \lambda_{i,r_i+1} = \ldots = \lambda_{i,J_i N_i} = 0. \)

Then the problem (P2i’) can be rewritten in a compact form as

\[
(P3i) : \min_{\tilde{f}_i} \tilde{f}_i^H \mathbf{M}_i \tilde{f}_i - 2 \text{Re} \{\mathbf{b}_i^H \tilde{f}_i\}; \tag{25a}
\]

s.t. \( \|\tilde{f}_i\|^2 \leq P_i. \tag{25b} \]

Since \( \mathbf{M}_i \) is positive semi-definite, (P3i) is convex. Also it is obvious that (P3i) is strictly feasible. Thus to solve problem
We now claim that the above equation (30) has a feasible solution. Indeed, this equation is solvable if and only if the right hand side \( b_i \) belongs to the column space \( \mathcal{R}(M_i) \).
Recall that \( M_i \) is Hermitian and has rank \( r_i \); so \( \mathcal{R}(M_i) = \text{span}(u_{i,1}, \ldots, u_{i,r_i}) \) and the null space of \( M_i \) satisfies \( \mathcal{N}(M_i) = \mathbb{R}^2(\tilde{M}_i) = \text{span}(u_{i,r_i+1}, \ldots, u_{i,K,N_i}) \). In fact, \( \mathbb{R}^{J_i,N_i} = \mathcal{R}(M_i) \oplus \mathcal{N}(M_i) \). Invoking the assumption of CASE (II) that \( |p_{i,k}| = 0, \forall k \in \{r_i+1, \ldots, J_i\} \) and the definition of \( p_i \), we obtain \( p_{i,k} = u_{i,k}^H b_i, \forall k \in \{r_i+1, \ldots, J_i\} \). Actually this implies \( b_i \in \mathbb{N}^2(\tilde{M}_i) = \mathcal{R}(\tilde{M}_i) \) and thus the consistency (i.e. the feasibility) of (30) is guaranteed.

Next we proceed to analytically identifying one feasible solution of (30). Taking eigenvalue decomposition on \( M_i \), (30) can be equivalently written as

\[
\Lambda_i^1 U_i^H \tilde{f}_i = p_i. 
\]  

Let \( \Lambda_i \) represent the top-left \( r_i \times r_i \) sub-matrix of \( \Lambda_i \), i.e. \( \Lambda_i = \text{diag}\{\Lambda_i^1, \mathbf{O}_{J_i-N_i-r_i}\} \). Let \( \bar{U}_i \) and \( U_i \) represent the left-most \( r_i \) columns and the remaining columns of \( U_i \) respectively, i.e. \( U_i = [\bar{U}_i, U_i] \). We can then simplify (31) to

\[
\Lambda_i^{\dagger} U_i^H \tilde{f}_i = p_i. 
\]  

Since the columns of \( U_i \) form a set of orthonormal basis for \( \mathbb{R}^{J_i,N_i} \), \( \bar{U}_i \) can be expressed via columns of \( U_i \) as

\[
\bar{U}_i = \sum_{k=1}^{J_i} \alpha_{i,k} u_{i,k}. 
\]  

Noting the key fact that \( \bar{U}_i^H u_{i,k} = 0, \forall k \in \{r_i+1, \ldots, J_i\} \), we know that the values of \( \{\alpha_{i,r_i+1}, \ldots, \alpha_{i,J_i}\} \) have no impact on (32) and can therefore be safely set to zeros to save energy. As for \( \alpha_{i,k}, \forall k \in \{1, \ldots, r_i\} \), we substitute \( \tilde{f}_i = \sum_{k=1}^{r_i} \alpha_{i,k} u_{i,k} \) into (32) and obtain

\[
\alpha_{i,k} = \lambda_{i,k}^{-1} p_{i,k}, \quad \forall k \in \{1, \ldots, r_i\}. 
\]  

Summarizing the above analysis, the optimal solution \( \tilde{f}_i \) to (P3i) is given by

\[
\tilde{f}_i = U_i \Lambda_i^1 U_i^H b_i, 
\]  

with \( \Lambda_i^1 \) being the Moore-Penrose pseudoinverse of \( \Lambda_i \) given as \( \text{diag}\{\Lambda_i^{-1}, \mathbf{O}_{J_i-N_i-r_i}\} \). Matrix Theory tells us that an arbitrary matrix \( X \) with its singular value decomposition (SVD) given by \( X = U_X \Lambda_X V_X^H \) has its unique Moore-Penrose pseudoinverse \( X^+ = V_X \Lambda_X^1 U_X^H \), where \( U_X \) and \( V_X \) are left and right singular square matrices, respectively, and \( \Lambda_X \) is a diagonal matrix with appropriate dimensions. Hence, (34) can be equivalently written as

\[
f_i^* = M_i^H b_i. 
\]  

Obviously \( \mu_i^* = 0 \), and \( f_i^* \) and \( \tilde{f}_i^* \) satisfy the KKT conditions (26a), (26c) and (26d). What remains to be shown is that \( \tilde{f}_i^* \) satisfies the power constraint. We verify this using (33) and get

\[
||\tilde{f}_i^*||^2 = \sum_{k=1}^{r_i} |\alpha_{i,k}|^2 \sum_{k=1}^{r_i} \frac{|p_{i,k}|^2}{\lambda_{i,k}^2} \leq P_i, 
\]  

(case i) if \( \sum_{k=r_i+1}^{J_i} |p_{i,k}|^2 = 0 \) and \( \sum_{k=1}^{r_i} \frac{|p_{i,k}|^2}{\lambda_{i,k}^2} > P_i \), a positive \( \mu_i^* \) satisfying (28) still exists and is unique. Consequently, the optimal solution \( \tilde{f}_i^* \) can be determined by (29) as in the subcase (i).
where the inequality follows from the assumption of CASE (II) again. Comparing it to the definition in (23a), we see that the solution in (23b) is actually identical to (23d).

Thus, we have thoroughly identified and written out the optimal solution to (P2′) for all the cases, which completes the proof.

We now provide several comments and supplementary discussion to help understand the structure of the solutions to (P2′) (or its compact form (P3i)).

Comment IV.1.

Algorithm 2: Bisection Search to Determine \( \mu^*_i \) in (28)

```plaintext
Input: \( \text{ubd, lbd, } g_i(\mu_i) \) in the form of (28), \( \delta \);
Output: \( \mu^*_i \) satisfying \( g_i(\mu^*_i) = 0 \);
1 Initialization: \( l = 1; \text{ubd}^{(0)} = \text{lbd}; \) ubd\(^{(0)}\) = ubd;
2 while \( (\text{ubd}^{(l)} - \text{lbd}^{(l)}) > \delta (\text{ubd}^{(l)} - \text{lbd}^{(l)}) \) do
3 \( l + 1; \alpha^{(l)} = \frac{\text{ubd}^{(l-1)} + \text{lbd}^{(l-1)}}{2} \);
4 if \( g_i(\alpha^{(l)}) \leq P_i \) then
5 \( \text{lbd}^{(l)} = \text{lbd}^{(l-1)}; \) ubd\(^{(l)}\) = \( \alpha^{(l)} \);
6 else
7 \( \text{lbd}^{(l)} = \alpha^{(l)}; \) ubd\(^{(l)}\) = \( \text{ubd}^{(l-1)} \);
8 end
9 Return \( \mu^*_i = \frac{\text{ubd}^{(l)} + \text{lbd}^{(l)}}{2} \);
```

In Algorithm 2, we provide a bisection search procedure to determine \( \mu^*_i \) in (28); \( \delta \) is a predefined parameter to indicate the relative precision. This is one efficient way to numerically determine the value of positive \( \mu^*_i \). Another choice is Newton’s method. Actually, solving (28) is equivalent to solving a secular problem, whose analytic solution is not available in [23]. It has been shown that the secular problem can be numerically solved with arbitrary precision by Newton’s method, which exhibits globally quadratic convergence rate and is usually faster than the bisection search method in [24] and [25]. However, Newton’s method involves the computation of the second-order derivatives. Further, in the context of (P2′), the complexity difference between the bisection search and Newton’s method is negligible compared to the dominant complexity coming from the matrix inversion operation (or pseudo-inversion) in (23a) and (23b).

Comment IV.2. When \( \mu^*_i = 0 \) and \( M_i \) is singular, the solution \( \mu^*_i \) to (30) is usually not unique. The solution provided in (23b) is the one with minimal transmission power among all the feasible solutions to (30) and guarantees to satisfy the power constraint.

Comment IV.3. It is worth noting that the three cases discussed in the proof of Theorem 3 - CASE(I)-case i), CASE(I)-case ii) and CASE(II), are mutually exclusive events. One and only one case will occur in given realization of (P2′).

Comment IV.4. Checking KKT conditions for each separate beamformer to obtain optimal solution is a rather standard approach, one that has been adopted in several previous studies such as [12], [13] and [8]. A big contribution here is that we have fully solved this problem by clearly identifying the solution structure and writing out the almost closed-form solutions for all possible cases, whereas the previous papers have not. Specifically, we have explicitly identify the conditions to determine the positiveness of \( \mu^*_i \) (three cases). For positive \( \mu^*_i \), we have identified a lower bound and an upper bound (in Lemma 2) on which numerical search can be performed. For the case of zero \( \mu^*_i \), one key consideration is the rank deficiency of \( M_i \). When \( M_i \) does not have a full rank, its inverse does not exist and consequently the solution given in [12], [13] and [8] is not applicable any more. Turns out, the rank deficiency scenario is actually not rare. In fact, whenever \( K < N_i \) or \( M < N_i \) holds, the matrices \( A_{ii} \) and \( C_i \) are born rank deficient. If they share common components of null space, \( M_i \) will be rank deficient. For example, consider the simple case where \( K_i = I_K, \Sigma_i = \sigma_i^2 I, \Sigma_i = \sigma_i^2 I \) and \( \min(K, M) < N_i \). Obviously \( M_i \) does not have a full rank. Thus considering the rank deficiency scenario of \( M_i \) is both meaningful and necessary. For the case of zero \( \mu^*_i \) and rank deficient \( M_i \), the solutions to the linear system (30) in the KKT conditions are not unique and can be infeasible to the power constraints. As shown in the proof, the solution in (23b) is the one that has the minimal transmission power and guarantees feasibility.

Comment IV.5. In the special case where \( K = J_i = 1 \), the fully closed form solution to (P2′) does exist! At this time, the value of positive \( \mu^*_i \) can be obtained analytically. Moreover, eigenvalue decomposition does not need to be invoked. So when \( K = J_i = 1 \), solving (P2′) is extremely efficient. For brevity, the detailed closed form solution for this special case is omitted in here.

Now what remains is to identify the lower bound \( \text{ubd}_i \) and the upper bound \( \text{ubd}_i \) for positive \( \mu^*_i \) in theorem 3 based on which the bisection search can be performed. The following Lemma provides the answer.

**Lemma 2.** The positive \( \mu^*_i \) in (P2′) (i.e. CASE (I) in Theorem 3) has the following lower bound \( \text{lbd}_i \) and upper bound \( \text{ubd}_i \):

i) For subcase i)

\[
\text{lbd}_i = \left[ \frac{\|p_i\|^2}{\lambda_{i,1}} \right]^+, \quad \text{ubd}_i = \left[ \frac{\|p_i\|^2}{\lambda_{i,1}} \right]^{-}
\]

ii) For subcase ii)

\[
\text{lbd}_i = \left[ \frac{\|p_i\|^2}{\lambda_{i,1}} \right]^+, \quad \text{ubd}_i = \left[ \frac{\|p_i\|^2}{\lambda_{i,1}} \right]^{-}
\]

**Proof:** For subcase i), by definition of \( g_i(\mu_i) \) in (28), we have

\[
\frac{\|p_i\|^2}{(\mu_i + \lambda_{i,1})^2} = \sum_{k=1}^{J_i} \frac{|p_{i,k}|^2}{(\mu_i + \lambda_{i,1})^2} \leq g_i(\mu_i) = P_i
\]

\[
\leq \sum_{k=1}^{J_i} \frac{|p_{i,k}|^2}{\mu_i^2} = \left[ \frac{\|p_i\|^2}{\mu_i^2} \right], \quad (39)
\]
which can be equivalently written as
\[
\frac{\|p_i\|^2}{\sqrt{P_i}} - \lambda_{i,1} \leq \mu_i \leq \frac{\|p_i\|^2}{\sqrt{P_i}}. \tag{40}
\]
Also notice that \( \mu_i^\star \) should be positive; the bounds in (37) thus follows.

For subcase ii), by assumption, \( \sum_{i,k=1}^{J,N_i} |p_{i,k}|^2 = 0 \). This leads to
\[
\frac{\|p_i\|^2}{(\mu_i + \lambda_{i,1})^2} = \frac{\sum_{k=1}^{r_i} |p_{i,k}|^2}{(\mu_i + \lambda_{i,1})^2} \leq g_i(\mu_i) = p_i \leq \frac{\sum_{k=1}^{r_i} |p_{i,k}|^2}{(\mu_i + \lambda_{i,r_i})^2} = \frac{\|p_i\|^2}{(\mu_i + \lambda_{i,r_i})^2}. \tag{41}
\]
Following the same line of derivation as in Subcase i), we obtain the bounds in (38).

Algorithm 3: Solving the Problem (P2')

Input: \( A_{ii}, C_i, B_i, q_i, \delta \);
Output: \( f_i^\star \);
1 Initialization: Perform eigenvalue decomposition \( M_i = U^H_iA_iU_i \); Calculate \( p_i \) using (24d);
2 if \( \exists k \in \{ r_i + 1, \ldots, J, N_i \} \) s.t. \( |p_{i,k}| \neq 0 \) then
3 Determine bounds \( \text{lbd}_i \) and \( \text{ubd}_i \) via (37):
4 Perform Algorithm 2 on \( [\text{lbd}_i, \text{ubd}_i] \) to determine \( \mu_i^\star \);
5 \( f_i^\star = (A_{ii} + C_i + \mu_i^\star E_i)^{-1}(B_i^H g - q_i) \);
6 else if \( \sum_{k=1}^{r_i} |p_{i,k}|^2 = 0 \) and \( \sum_{k=1}^{r_i} |p_{i,k}|^2 > P_i \) then
7 Determine bounds \( \text{lbd}_i \) and \( \text{ubd}_i \) via (38):
8 Perform Algorithm 2 on \( [\text{lbd}_i, \text{ubd}_i] \) to determine \( \mu_i^\star \);
9 \( f_i^\star = (A_{ii} + C_i + \mu_i^\star E_i)^{-1}(B_i^H g - q_i) \);
10 else
11 \( f_i^\star = E_i^{-\frac{1}{2}} (E_i^{-\frac{1}{2}} (A_{ii} + C_i) E_i^{-\frac{1}{2}})^{-1} E_i^{-\frac{1}{2}} (B_i^H g - q_i) \);
12 end
13 Return \( f_i^\star \);

Algorithm 3 summarizes the results obtained in Theorem 3 and Lemma 2 to provide a BCD-based (nearly) closed-form solution to (P2'). Here, \( \delta \) is a predefined threshold to control the accuracy of the bisection search.

B. Layered-BCD Algorithm

The above analysis of (P2'), combined with (P1), naturally leads to a nested or layered BCD algorithm, that can be used to analytically solve the joint beamforming problem (P0). The algorithm consists of two layers (two loops). The outer layer is a two-block descent procedure alternatively optimizing \( G \) and \( \{F_i\}_i \), and the inner loop further decomposes the optimization of \( \{F_i\}_i \) into an L-block descent procedure operated in an iterative round robin fashion. Algorithm 4 outlines the overall procedure.

Next we analyze the convergence of this layered-BCD method.

Lemma 3 (Proposition 6 in [18]). Consider the problem set up in (79) and the assumption therein. Suppose that function \( f(\cdot) \) is pseudoconvex on \( X \) and there exists some point \( \bar{x} \in X \) such that the level set \( \{ x \in X | f(x) \leq f(\bar{x}) \} \) is compact. The solution sequence \( \{x^{(k)}\} \) is generated by block coordinate descent method. Then every limit point of \( \{x^{(k)}\} \) is a stationary point of the problem (79).

Theorem 4. The objective sequence \( \{MSE^{(j)}\}_{j=0}^\infty \) generated by Algorithm 2 is monotonically decreasing. By setting \( \epsilon_0 \to 0 \) (or \( M_0 \to \infty \)) and \( \epsilon_1 \to 0 \) (or \( M_1 \to \infty \)), the solution sequence \( \{\{F_i^{(j)}\}_i\}_{i=1}^\infty \), \( \{G^{(j)}\}_i \) generated by Algorithm 2 has limit points, and every limit point of \( \{\{F_i^{(j)}\}_i\}_{i=1}^\infty \), \( G^{(j)} \) is a stationary point of (P0).

Proof: The proof of the monotonicity of \( \{MSE^{(j)}\}_{j=0}^\infty \) and the existence of limit points for the solution sequence \( \{\{F_i^{(j)}\}_i\}_{i=1}^\infty \), \( G^{(j)} \) follows the same lines of thoughts as that of Theorem 2.

From Theorem 1 given \( g \), the objective function \( MSE(\{f_i^{(j)}\}_{j=1}^\infty) \) of Problem 172 is convex (and so of course pseudoconvex) with respect to \( \{f_i^{(j)}\}_{j=1}^\infty \). Since the objective \( MSE(\{f_i^{(j)}\}_{j=1}^\infty) \) in (P2') (or (P2) equivalently) is continuous and the feasible domain for \( \{f_i^{(j)}\}_{j=1}^\infty \) is bounded, there exist feasible points \( \{f_i^{(j)}\}_{j=1}^\infty \) making the solution set
\[
\{f_i\}_{i=1}^k \in \mathbb{C}^{\mathbb{N}_1 \times 1} \times \cdots \times \mathbb{C}^{\mathbb{N}_L \times 1} | \text{MSE}\{f_i\}_{i=1}^L | g \leq \\
\text{MSE}\{f_i\}_{i=1}^L | g \}
\]
closed and bounded. Thus by Lemma 3, when \(\epsilon_1 \to 0\) (or \(M_1 \to \infty\)), for any \(j \in \mathbb{N}_+\) with \(g^{(j)}\), there exists some subsequence \(\{f_i^{(j)}\}_{i=1}^\infty\) converging to a stationary point of the problem (P2) associated with \(g^{(j)}\). Since (P2) is a convex problem, any stationary point is actually a globally optimal solution. Thus, when \(\epsilon_1 \to 0\) (or \(M_1 \to \infty\)), the subproblem (P2) is actually globally solved (given \(g^{(j)}\)). Finally, from Theorem 2 when \(\epsilon_0 \to 0\) (or \(M_0 \to \infty\)), each limit point of \(\{f_i^{(j)}\}_{i=1}^L, G^{(j)}\}_{j=1}^\infty\) is a stationary point of of the original problem (P0).

Theorem 5. The objective sequence \(\{\text{MSE}^{(j)}\}_{j=0}^\infty\) generated by Algorithm 4 with a finite \(M_1\) is monotonically decreasing. By setting \(\epsilon_0 \to 0\) (or \(M_0 \to \infty\)), the solution sequence \(\{f_i^{(j)}\}_{i=1}^L, G^{(j)}\}_{j=1}^\infty\) has limit point(s). If the entire sequence \(\{f_i^{(j)}\}_{i=1}^L, G^{(j)}\}_{j=1}^\infty\) converges, then its limit is a stationary point of (P0).

Proof: The proof for the monotonicity and the existence of the limit point(s) is similar to that of Theorem 4 and Theorem 2 and is therefore omitted.

Assume that the sequence \(\{f_i^{(j)}\}_{i=1}^L, G^{(j)}\}_{j=1}^\infty\) converges to the limit point \(X \triangleq \{f_i^{(j)}\}_{i=1}^L, G\), we now show that this limit point is also stationary.

Since the update of \(G^{(j+1)}\) solves an optimization problem with \(f_i^{(j)}\), as a necessary condition of optimality of \(G^{(j+1)}\), we have \(\text{Tr}\{(\nabla_{G^{(j+1)}} \text{MSE}(G^{(j)}))_{i=1}^L, G^{(j+1)}(G - G^{(j+1)})\} \geq 0\) for any \(G \in \mathbb{C}^{M \times K}\). Since \(f_i^{(j)} \rightarrow f_i^{(j+1)}\) and \(G^{(j+1)} \rightarrow G\) and \(\nabla_{G} \text{MSE}\) is continuous in \(G\) and \(f_i^{(j)}\), we have \(\text{Tr}\{(\nabla_{G^{(j+1)}} \text{MSE}(X))(G - G^{(j+1)})\} \geq 0\) for any \(G \in \mathbb{C}^{M \times K}\). The same reasoning suggests that \(\text{Tr}\{(\nabla_X \text{MSE}(X))_i, G^{(j)} - f_i^{(j)}\}_i \geq 0\) for all feasible \(f_i\), \(i = 1, \cdots, L\). By summing up these \(L\) + 1 terms, we have \(\text{Tr}\{(\nabla_X \text{MSE}(X))_i, G^{(j)} - f_i^{(j)}\}_i \geq 0\) for any \(X\) that is feasible for the original problem (P0). This suggests that the convergent limit point \(\{f_i^{(j)}\}_{i=1}^L, G\) is actually a stationary point of (P0).

It is worth noting that for the case of small \(M_1\), we have somewhat abused the name of layered-BCD algorithm. This is because, with a small \(M_1\), the subproblem (P2) cannot guarantee to be optimally solved and thus the outer layer is, strictly speaking, not a BCD algorithm any more. Nevertheless, for any finite \(M_1\), Algorithm 4 can still be regarded as a kind of “essentially cyclic BCD algorithm”, which is discussed in the next subsection.

Comment IV.6. The iterative algorithm proposed in [8] is actually a special case of our Algorithm 4 with \(N_1 = 1\). The solution provided in [8] ignored the case when \(\mu_i = 0\) and \(M_i\) is singular (and hence not invertible), and we have made it complete. We also provide a stronger convergence conclusion in Theorems 4 and 5 than that discussed in [8].

C. Cyclic (L+1)-BCD Algorithm and Its Approximation

In this subsection, we simplify the layered-BCD method discussed in the previous subsection by proposing an essentially cyclic BCD optimization method.

Generally speaking, to ensure the convergence of BCD methods for a problem with \(n\) blocks requires each block be updated sufficiently frequently. To fulfill this requirement, a commonly-used updating rule called essentially cyclic rule suggests finding a constant \(m \geq n\), such that each block is updated at least once within any consecutive \(m\) updating blocks \([16] [17]\). A particularly popular case of essentially cyclic rule is to have \(m = n\), which ensures that all the blocks are updated in a Round Robin manner.

Here we decompose the variables of (P0) into \((L + 1)\) blocks: \(G\) and \(F_i\)’s for \(i = 1, 2, \cdots, L\). The update rule is set such that \(F_i\)’s (for \(i = 1, 2, \cdots, L\)) will be updated successively, and the update of each \(F_i\) will be followed immediately by a calibration of \(G\). Hence in each period of \(m = 2L\) iterations, we have the following update sequence:

\[ \cdots, F_1, G, F_2, G, \cdots, \]

This \((L + 1)\) block essentially cyclic optimization is detailed in Algorithm 5.

**Algorithm 5: Cyclic BCD Algorithm to Solve (P0)**

**Input:** \(\Sigma, \{\Sigma_i\}_{i=1}^L, \{K_i\}_{i=1}^L, \{H_i\}_{i=1}^L, \{P_i\}_{i=1}^L, \epsilon_0 (or M_0); \)

**Output:** \(\{F_i\}_{i=1}^L\) and \(G\)

1. **Initialization:** randomly generate feasible \(\{F_i\}_{i=1}^L; \)
2. obtain \(G^{(0, L)}\) by [10]: \(\text{MSE}(-1) = \infty; \)
3. calculate \(\text{MSE}^{(0)} = \text{MSE}(\{F_i^{(0)}\}_{i=1}^L, G^{(0, L)})\) by [7];
4. for \(j = 1; \text{MSE}^{(j-2)} - \text{MSE}^{(j-1)} > \epsilon_0 (or J < M_0); j++ \) do
5. for \(i = 1; i <= L; i++ \) do
6. with \(G^{(j-1, i)} = \{F_k^{(j-1, i)}\}_{k=1}^{i-1} \) and \(\{F_k^{(j-1, i)}\}_{k=i+1}^L\) being fixed;
7. invoke Alg 3 to obtain \(F^{(j)}\);
8. update \(G^{(j, i)}\) by [10] with \(\{F_k^{(j)}\}_{k=1}^{i} \) and \(\{F_k^{(j-1, i)}\}_{k=i+1}^L\) given;
9. \(k++\);
10. end
11. end
12. calculate \(\text{MSE}^{(j)} = \text{MSE}(\{F_i^{(j)}\}_{i=1}^L, G^{(j, L)})\) by [7];
13. return \(\{F_i^{(j)}\}_{i=1}^L, G^{(j, L)}\);

The convergence properties of this cyclic \((L + 1)\)-BCD algorithm can be analyzed in ways similar to the previous
algorithms (e.g. Theorem 4). To save space, here we omit the detailed proof, and only present the convergence results in Theorem 6.

**Theorem 6.** The objective sequence \( \{\text{MSE}^{(j)}\}_{j=0}^{\infty} \) generated by Algorithm 5 is monotonically decreasing. By setting \( \epsilon_0 \to 0 \) (or \( N_0 \to \infty \)), the solution sequence \( \{(F_{i}^{(j)}), L_{i=1}^{\infty}, G^{(j)}\}_{j=1}^{\infty} \) has limit point(s). If the sequence \( \{(F_{i}^{(j)}), L_{i=1}^{\infty}, G^{(j)}\}_{j=1}^{\infty} \) converges, then its limit is a stationary point of \((P0)\).

**Comment IV.7.** Although Theorem 6 is presented for the proposed Algorithm 5, the converging results stated therein actually apply to any essentially cyclic order, and not necessarily the “interleaved” order proposed here.

**Comment IV.8.** It should be noted that although Theorem 5 and Theorem 6 have proven that the corresponding solution sequences have limit point(s), unlike Theorem 4 and Theorem 3, it is hard to prove that any of these limit point(s) is stationary. Generally speaking, for essential cyclic BCD algorithms, the existence of the limit points together with the solvability of each block subproblem are not sufficient to guarantee that these limit points are stationary. In fact, there exist counterexamples [18] and [19], where each subproblem can be optimally solved, but the solution sequence has only non-stationary limit points. Usually convergence to stationary points requires more assumptions or constraints on the problem structure. To ensure limit points are also stationary, for example, [14] assumes convexity of the whole problem and strict convexity of each subproblem; [15] assumes uniqueness of optimal solution to each subproblem and a special monotonicity in each block coordinate; [18] assumes strict quasi-convexity in each block coordinate or pseudo-convexity of the whole problem, and compactness of the level set. In our problems, we have assumed the convergence of the entire solution sequence in Theorems 5 and 6, in order to show that the limit points are also stationary. However, that during our extensive simulation experiments, we have observed that Algorithm 5 always gives out convergent solution sequence. In other words, Theorems 5 and 6 are good enough to convey the convergence natures of algorithms 5 and 4 with finite inner iteration.

**D. An Approximated Cyclic \((L + 1)\)-BCD Algorithm**

It is also worth noting that we can accelerate the aforementioned essentially cyclic BCD optimization in Algorithm 5 by introducing an approximation as follows: When solving the subproblem \((P2')\) to update \( F_i \), in addition to setting the other \( \{F_j\}_{j \neq i} \) being known and fixed, we can assume that the term \( A_{i}f_{i} \) is also known by leveraging the value of \( f_{i} \) in the previous update iteration. In other words, we assume \( q_{i} = \sum_{j=1}^{L} A_{ij} f_{j} \) is given and thus the matrix \( M_{i} \) in \((P2')\) is actually \( C_{i} \). Interesting enough, our simulation experiments show that this approximation actually also significantly improves the convergence rate of the cyclic-BCD procedure (in addition to slight complexity saving in each iteration)!

**V. Numerical Results**

We now present simulation results to verify the efficiency of the proposed BCD optimization algorithms and the effectiveness of the resulting beamformers. We consider a wireless sensor network with \( L = 3 \) sensors and a fusion center, all observation matrices \( K_i \) at the sensors are identity matrices, and that all the nodes are equipped with 3 antennae, i.e. \( N_1 = N_2 = N_3 = M = 3 \). We set the transmit power for each of the three sensors to be \( P_1 = 2 \), \( P_2 = 2 \) and \( P_3 = 3 \), respectively. The source signal \( s \) is assumed to have a vector size \( K = 3 \) (i.e. rate-1 beamforming), unit-power per dimension, and zero mean and white covariance. The noise at each sensor is white and set at the levels of \( SNR_1 = 5 \text{ dB}, \ SNR_2 = 10 \text{ dB} \) and \( SNR_3 = 8 \text{ dB} \), respectively.

First, we examine the finding in Subsection IV.A decomposes \( \{F_i\}_{i=1}^{L} \) into \( L \) blocks of individual \( F_i \) can globally optimize the subproblem \((P2')\). This involves the repeated run of Algorithm 5 from \( i = 1 \) to \( L \), and back again, and again. We have randomly generated 1000 channel profiles with channel SNR covering the range of \( SNR_0 = 2 \text{ dB} \) to 10\text{dB}, and each channel profile involves a randomly generated channel matrices \( \{H_1, H_2, H_3\} \). Since the postcoder \( G \) is assumed known and irrelevant to the optimization of the precoders, its realization is also randomly chosen. For each channel realization, we first reformulate the problem \((P2')\) to a convex optimization problem as in [77], and apply the renowned tool CVX to get the solution. We then perform the decomposed BCD method \((\text{Algorithm 3})\) with some random initial point satisfying all the power constraints [59]. In all of the tests, we observe the same global convergence effect. For clarity, only the results of 10 channel profiles are plotted in Fig 2. The optimal values obtained by the CVX solver serve as the benchmarks (the flat horizontal lines). We see that the decomposed BCD method generally reaches the optimal value within 20 iterations (each iteration involves the the successive update of every \( F_i \) once). This confirms our conclusion that, given sufficiently large number of inner iteration, \((P2')\) can be optimally solved and consequently the layered-BCD algorithm can converge to stationary points.

Next, we test and compare the performance and the complexity of the proposed BCD-based algorithms. We evaluate four different types: 2-BCD \((\text{Algorithm 1})\), layered-BCD with inner iteration set to 2 \((\text{Algorithm 3})\), cyclic \((L+1)\)-BCD \((\text{Algorithm 5})\), and the approximated version of cyclic \((L+1)\)-BCD. In these simulation tests, we let the channel noise level increases from \( SNR_0 = 2 \text{ dB} \) to 20\text{dB}. At each specific channel noise level, 1000 channel realizations \( \{H_1, H_2, H_3\} \) are randomly generated, each of whose elements follows standard complex circular Gaussian distribution \( \mathcal{CN}(0, 1) \). The mean square error averaged over these 1000 random channel realizations are evaluated as a function of the number of (outer) iterations (and the channel SNR).

We present in Table II the MATLAB running time for there four different algorithms (running in a regular laptop). For simplicity, we focus on the rate-1 beamforming case, and
Table I: MATLAB Running Time Per (Outer) Iteration

| Dim. | Algorithms          | $L = 2$ | $L = 4$ | $L = 6$ | $L = 8$ |
|------|---------------------|---------|---------|---------|---------|
| $K = 3$ | 2-BCD               | 0.2107s | 0.2390s | 0.2987s | 0.3500s |
| $N_i = 3$ | Cyc. BCD       | 0.0026s | 0.0066s | 0.0120s | 0.0189s |
| $K = 3$ | 2-BCD               | 0.2432s | 0.3068s | 0.3636s | 0.4285s |
| $M = 3$ | Cyc. BCD           | 0.0086s | 0.0159s | 0.0328s | 0.0560s |
| $N_i = 2$ | Lay. BCD       | 0.0087s | 0.0241s | 0.0493s | 0.0839s |
| $K = 6$ | 2-BCD               | 0.2529s | 0.3786s | 0.5801s | 0.7520s |
| $M = 6$ | Cyc. BCD           | 0.0307s | 0.0505s | 0.0997s | 0.1594s |
| $N_i = 2$ | Lay. BCD       | 0.0116s | 0.0219s | 0.0622s | 0.1031s |
| $K = 9$ | 2-BCD               | 0.4352s | 0.7956s | 1.1401s | 1.9593s |
| $M = 9$ | Cyc. BCD           | 0.0120s | 0.0302s | 0.0557s | 0.0902s |
| $N_i = 2$ | Lay. BCD        | 0.0208s | 0.0514s | 0.0928s | 0.1461s |

Note: (i) layered-BCD is run with 2 inner iterations only.
(ii) approximate cyclic-BCD has essentially the same running-time as non-approximate cyclic-BCD.

The MSE and the convergence performance of the resulting beamformers are illustrated in Figures 3–8. We note that layered-BCD (with 2 inner iterations) and cyclic $(L+1)$-BCD have rather similar performances, with the latter having slightly lower complexity and slightly slower convergence than the former. Putting them together in the same figure causes very cluttered plots that are hard to read. Hence, for clarity purpose, and for the sake of demonstrating optimality, we use 2-BCD as the benchmark (CVX solver gives optimal solution to (P2)), and compare each one of the other algorithms against it.

Figure 3 compares layered-BCD with 2-BCD. From Fig. 2 we know that 2 inner iterations are hardly sufficient to achieve globally optimal solution to the subproblem (P2). However, as the outer iteration number increases, the deficiency of the inner iteration is quickly made up for, and the algorithm almost always converges after 30 iterations. Since one iteration of 2-BCD take as much running time as 20 (outer) iterations of layered-BCD (with $L = N_i = M = K = 3$), 30-layered-BCD iterations would amount to less than 2 2-BCD iterations, and clearly the former lavishly out-wins the latter.

Figure 4 compares cyclic $(L+1)$-BCD with 2-BCD. Similar to the layered-BCD case, cyclic-BCD achieves the same good performance as 2-BCD after some 30 iterations, but consumes only a tiny fraction of the complexity required by the latter.

Figure 5 compares approximate cyclic $(L+1)$-BCD with 2-BCD. The performance is unexpectedly impressive. As shown in the figure (repeated tested and averaged over many runs), the approximate version can exhibit excellent performance within less than 5 iterations and, with some 10 iterations, it produce such superb results that even 2-BCD is incapable of. Although we are unable to explain why, this simulation demonstration is very fascinating, especially considering the very low complexity of the approximated algorithm.

Next, we assess the convergence properties of these algorithms. We set $\text{SNR}_0 = 8$ dB and fix the channel with $K = M = N_i = J_i$. Besides, the special case of $K = 1 = J_i$ is also tested. Different values of $K$ (size of the source vector) and $L$ (number of sensors) are evaluated to take into account the influence of the problem dimensions. For each randomly-generated channel realization and initial point, we let each algorithm run 50 (outer) iterations.

As discussed previously, layered-BCD may be viewed as a special type of essentially cyclic BCD, and hence can expect to eventually exhibit similar performance.
a randomly-generated realization. We randomly generate 20 initial points which satisfy the power constraints. The proposed algorithms run from these initial points and the resultant MSE itineraries are plotted in Figures 6-8. These plots clearly demonstrate that all the proposed algorithms are insensitive to initial points and would exhibit rather stable convergence. With a large number of iterations (e.g., 30 or above), we see that layered-BCD and cyclic \((L+1)-\text{BCD}\) tend to outperform 2-BCD. Interestingly, from Figure 5, we find that cyclic \((L+1)-\text{BCD}\) algorithm with approximation does not always guarantee monotonically decreasing MSE value, but the fluctuation seems to occur only at a very early state (such as in the first 3 iterations). Beyond that level, the approximate method does exhibit an surprisingly fast convergence and a superb MSE value (as verified by extensive simulation tests).

VI. CONCLUSION

We have studied the linear beamforming design problem for a centralized wireless sensor network under the MSE criterion. We consider a very general model which allows for individually contaminated source and separate power constraint for each sensor, and whose solution still rather incomplete.

Since the joint beamformer design problem is nonconvex, we propose to approach it using the block coordinate decent method. A two-block BCD algorithm is first proposed, which decomposes the original problem into two subproblems that alternatively optimize the linear postcoder and the linear precoders (with the other being fixed). We show that the postcoder...
can be analytically solved using the Wiener filter, and that the precoders collectively constitute a convex optimization problem (and can therefore be solved using existing tools such as CVX). Based on our analysis, we further decompose the problem of optimizing the set of precoders into multiple blocks, each targeting solving a single precoder. Two specific BCD forms, termed layered-BCD and cyclic-BCD, are discussed in detail. The convergence properties of these proposed algorithms are carefully analyzed. Extensive numerical results confirm the efficiency of the proposed algorithms.

References

[1] L. Xiao and S. Boyd, “Fast linear iterations for distributed averaging,” Syst. Control Lett., vol. 53, no. 1, pp. 65-78, 2004.

[2] A. Bertrand and M. Moonen, “Distributed adaptive node-specific signal estimation in fully connected sensor networkpart I: Sequential node updating,” IEEE Trans. Signal Process., vol. 58, pp. 5277-5291, 2010.

[3] A. Bertrand and M. Moonen, “Distributed adaptive node-specific signal estimation in fully connected sensor networkpart II: Simultaneous and asynchronous node updating,” IEEE Trans. Signal Process., vol. 58, pp. 5292-5306, 2010.

[4] Y. Zhu, E. Song, J. Zhou, and Z. You, “Optimal dimensionality reduction of sensor data in multisensor estimation fusion,” IEEE Trans. Signal Process., vol. 53, no. 5, pp. 1631-1639, May 2005.

[5] J. Fang and H. Li, “Optimal/near-optimal dimensionality reduction for distributed estimation in homogeneous and certain inhomogeneous scenarios,” IEEE Trans. Signal Process., vol. 58, no. 8, pp. 4339-4353, 2010.

[6] I. D. Schizas, G. B. Giannakis, and Z.-Q. Luo, “Distributed estimation using reduced-dimensionality sensor observations,” IEEE Trans. Signal Process., vol. 55, no. 8, pp. 4284-4299, Aug. 2007.

[7] J. Xiao, S. Cui, Z. Luo, and A. J. Goldsmith, “Linear coherent decentralized estimation,” IEEE Trans. Signal Process., vol. 56, no. 2, pp. 757-770, Feb. 2008.

[8] A. S. Bebbahani, A. M. Eltawil, H. Jafarkhani, “Linear Decentralized Estimation of Correlated Data for Power-Constrained Wireless Sensor Networks,” IEEE Trans. Signal Process., vol. 60, no. 11, pp. 6003-6016, Nov. 2012.

[9] W. Guan and H. Luo, “Joint MMSE transceiver design in nonregenerative MIMO relay systems,” IEEE Commun. Lett., vol. 12, pp. 517-519, July 2008.

[10] Y. Rong, X. Tang, and Y. Hua, “A unified framework for optimizing linear non-regenerative multicarrier MIMO Relay Communication Systems,” IEEE Trans. Signal Processing, vol. 57, pp. 4837-4851, Dec. 2009.

[11] Y. Rong, “Optimal joint source and relay beamforming for MIMO relays with direct link,” IEEE Commun. Lett., vol. 14, pp. 390-392, May 2010.

[12] S. Serbetli and A. Yener, “Transceiver optimization for multiuser MIMO systems,” IEEE Trans. Signal Process., vol. 52, pp. 214-226, Jan. 2004.

[13] M. R. A. Khandaker and Y. Rong, “Joint transceiver Optimization for multiuser MIMO relay communication systems”, IEEE Trans. Signal Processing, vol. 60, pp. 5977-5986, Nov. 2012.

[14] D. Bertsekas and J. Tsitsiklis, Parallel and Distributed Computation, Englewood Cliffs, NJ: Prentice-Hall, 1989.

[15] D. Bertsekas, Nonlinear Programming, 2nd ed., Belmont, MA:Athena Scientific, 1999.

[16] P. Tseng, “Convergence of a block coordinate descent method for nondifferentiable minimization,” Journ. of Optim. Theory and Appl., vol. 109, no. 2, pp. 475-494, June 2001.

[17] P. Tseng, “Dual ascent methods for problems with strictly convex consis and linear constraints: a unified approach,” SIAM Journ. on Contr. and Optim., vol. 28, pp. 214-242, 1990.

[18] L. Grippo and M. Sciandrone, “On the convergence of the block nonlinear Gauss-Seidel method under convex constraints,” Operations Research Letters, vol. 26, pp. 127-136, 2000.

[19] M.J.D. Powell, “On search direction for minimization algorithms,” Math. Programming, vol. 4, no. 1, pp. 193-201 1973.

[20] T. P. Minka. Old and New Matrix Algebra Useful for Statistics, Notes, December 2000.

[21] S. Haykin, Adaptive Filter Theory (4th ed.), Upper Saddle River, NJ: Prentice-Hall, 2002.

[22] S. Boyd, and L. Vandenberghe, Convex Optimization. New York: Cambridge University Press, 2004.

[23] M. Grant and S. Boyd, “CVX: Matlab software for disciplined convex programming (web page and software),” [Online]. Available: http://cvxr.com/cvx, Apr. 2010.

[24] A. Melman, “A unifying convergence analysis of second-order methods for secular equations,” Math. Comp., vol.66, pp. 333-344, 1997.

[25] A. Beck, A. Ben-Tal, and M. Teboulle, “Finding a global optimal solution for a quadratically constrained fractional quadratic problem with applications to the regularized total least squares, SIAM J. Matrix Anal. Appl., vol. 28, no. 2, pp. 425-445, 2006.