

Maximin Joint Optimization of Transmitting Code and Receiving Filter in Radar and Communications

Licheng Zhao and Daniel P. Palomar, *Fellow, IEEE*

Abstract—In this paper, we conduct the joint design of transmitting sequence(s) and receiving filters subject to the Peak-to-Average Ratio (PAR) constraint in radar and communications applications. We consider optimizing the worst-case performance and the resulting optimization problem takes a maximin format. We propose two algorithms based on the MM (Majorization–Minimization or Minorization–Maximization) method as opposed to the traditional epigraph-based smooth reformulation. On top of that, both algorithms are guaranteed to converge to a B(oulingand)-stationary solution, and B-stationarity is the appropriate stationarity condition for problems with a nonconvex constraint set. The proposed algorithms successively solve a series of simple convex problems that enjoy low computational complexity. Numerical simulations have shown that the proposed algorithms empirically achieve slightly higher objective values and converge faster in terms of CPU time than the existing methods.

Index Terms—Radar, CDMA, PAR constraint, maximin, MM method, B-stationary solution.

I. INTRODUCTION

MAXIMIZATION of the minimum of a finite number of differentiable functions is of interest in various signal processing applications, especially in radar target detection and multiuser communications. The maximin metric aims at ensuring the worst-case performance guarantee or providing fairness among multiple users. In this paper, we consider the maximization of the minimum of several Signal-to-Interference-plus-Noise Ratio (SINR) functions subject to the PAR constraint, i.e.,

$$\begin{aligned} & \underset{\mathbf{s} \text{ (or } \mathbf{s}_i), \mathbf{w}_i}{\text{maximize}} && \min_{i=1,2,\dots,I} \text{SINR}_i \\ & \text{subject to} && \mathbf{s} \text{ (or } \mathbf{s}_i) \in \mathcal{S}, \end{aligned} \quad (1)$$

where \mathbf{s} (or \mathbf{s}_i) $\in \mathbb{C}^N$ denotes the transmitting sequence, $\mathbf{w}_i \in \mathbb{C}^M$ represents the receiving filter, and \mathcal{S} models the PAR constraint set (cf. [1]–[4]):

$$\mathcal{S} = \left\{ \mathbf{s} \in \mathbb{C}^N \mid \|\mathbf{s}\|_2 = 1, \|\mathbf{s}\|_\infty \leq \sqrt{\frac{\rho}{N}} \right\}. \quad (2)$$

The PAR constraint controls the excursions of the squared code elements around their mean value [3]. A lower PAR means a lower dynamic range of the analog-to-digital converters and

digital-to-analog converters in the system, and fewer linear power amplifiers are needed. The ℓ_2 -norm constraint stands for the energy budget and the ℓ_∞ -norm constraint reflects the PAR level which is controlled by the parameter ρ , ranging from 1 to N . In particular, when $\rho = 1$, the PAR constraint degenerates into the constant modulus constraint. The different expressions of SINR_i will be specified in the next section.

A. Related Works

Radar target detection: Joint design of the receiving filter and transmitting sequence has been extensively studied during the last few decades. In the field of active sensing, many works are based on either known Doppler shifts [5]–[7] or signal-independent interference [3], [8], [9]. In practice, Doppler shifts are often unknown, especially when the detection process has just been launched and the target has not yet been tracked. The assumption of signal-independent interference fails to take into account possible reflections of transmitting signals from other objects (hence signal-dependent interference). One pioneering work combining these two considerations is [10], which proposed a novel algorithm, DESIDE, to conduct the maximin optimization. The DESIDE algorithm is cyclic and Semidefinite-Programming (SDP)-relaxation-based with overall complexity $\mathcal{O}(N^{6.5})$ [11], [12], and the computational cost is rather high. The more recent work [13] improved the design of [10] by incorporating a filter bank, i.e., multiple filters instead of one, on the receiver side. Each filter is tuned to a specific Doppler frequency, and all the Doppler frequencies are uniformly sampled from the uncertainty interval of the target Doppler frequency. The idea of filter banks originates from the Moving Target Detector (MTD) [14]. Incorporating the filter bank proves to enhance the worst-case performance according to the simulation results of [13]. Moreover, the algorithm proposed by [13] is Second-Order-Cone-Programming (SOCP)-based, with overall complexity $\mathcal{O}(N^{3.5})$ [11], [15], three orders of magnitude less costly than SDP. However, [13] merely solved a related problem rather than the original one (i.e., (1)), and the stationarity convergence result is thus not intended for the original problem.

Apart from active sensing, there exists a similar problem in a colocated MIMO radar system [16]–[19]. We consider signal-dependent interference as well as some uncertainty in the target angle. A filter bank is also used on the receiver side. Each filter is tuned to a specific predetermined target angle, and all the target angles are uniformly sampled from the uncertainty interval. Hence, the optimization problem takes the same form as (1).

Multiuser Communications: DS-CDMA: The user performance of Direct Sequence-Code Division Multiple Access (DS-CDMA) also depends on the joint design of receiving filters and transmitting sequences (also known as signature codes). To ensure max-min fairness among users, we maximize the

Manuscript received May 22, 2016; revised August 18, 2016 and October 7, 2016; accepted October 23, 2016. Date of publication November 4, 2016; date of current version December 5, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ami Wiesel. This work was supported in part by the Hong Kong RGC 16206315 research grant and in part by the Hong Kong RGC Theme-Based Research Scheme under Grant T21-602/15R.

The authors are with the Hong Kong University of Science and Technology, Hong Kong (e-mail: lzhaoui@ust.hk; palomar@ust.hk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSP.2016.2625267

minimum SINR of all users. Interestingly, as [20] pointed out, maximizing the minimum SINR is equivalent to maximizing the minimum largest achievable rate ($R = \log_2(1 + \text{SINR})$) as well as minimizing the maximum Mean Square Error (MSE) ($\text{MSE} = \frac{1}{1 + \text{SINR}}$). The expression of the SINR follows [21], and interference is caused by the signals of other users. The PAR constraint is imposed to prevent high peak power of linear combinations of signature sequences [22], [23]. Apart from DS-CDMA, there are many relevant works studying the maximin SINR problem. The works [24] and [25] studied the maximin SINR problem in a MIMO downlink system with [24] optimizing the transmitting beamformers only and [25] jointly designing the transmitting and receiving beamformers. Karipidis *et al.* [26] studied the max-min fair transmitting beamformers subject to quality of service constraints. Soltanalian *et al.* [27] proposed the Grab-n-Pull algorithm to design precoding vectors. Wu *et al.* [28] exploited semidefinite relaxation to design the transmitting matrix for relay beamforming networks. These works provide insight into solving the maximin SINR problem.

Algorithmic Scheme: From the perspective of the algorithm, maximin optimization has been well studied in the literature. Introducing a slack variable and deriving the equivalent epigraph-based reformulation is a classic and common practice [29]. One recent work by Scutari *et al.* [30] showed that the stationary solution of the epigraph-based reformulation turns out to be the d(irectional)-stationary solution of the original problem. However, when the constraint set includes a nonconvex equality constraint, the algorithm in [30] may fail and the convergence result is no longer applicable. Hence, this paper avoids the epigraph-based reformulation and works on the piecewise differentiable objective directly.

Orthogonality Concern: In sequence design, we could either impose mutual orthogonality of sequences in a direct way or in an indirect way. The direct way is to explicitly suppress the magnitude of the inner product of multiple sequences, like [2], [31]. This kind of design is useful when we do not have any prior knowledge of channel information. However, explicit orthogonality may not be necessary if we know in advance that some particular channel lags do not exist. We could take into account the channel information and maximize the worst-case SINR on the receiver side instead. By including the channel information, we are implicitly inducing mutual orthogonality, like [10], [13], [32]. In this paper, we exploit the information of channel and/or clutter in the sequence design and thus we do not adopt the explicit orthogonality philosophy.

B. Contribution

In this paper, we propose two algorithms based on the MM method to efficiently solve the maximin problem. The major contributions are as follows:

- 1) We employ the MM method and extend it to the case where the objective takes the pointwise minimum format. The tight lower bound for a piecewise smooth function is simple; however, it is nontrivial to verify the condition for stationarity convergence. We are able to claim convergence to a B(oulingand)-stationary solution of the original problem,¹ even if the objective function is only piecewise

smooth and the constraint set includes a nonconvex equality constraint, which is beyond the scope of [30].

- 2) The proposed algorithms can achieve slightly higher objective values empirically and are more efficient than the existing methods. We break the convention of alternating optimization in the joint design of receiving filters and transmitting sequence(s). The alternating algorithmic scheme gives rise to either SOCP- or SDP-based algorithms, which are computationally costly and often utilize an off-the-shelf solver. However, the proposed MM-based algorithms are more systematic and efficient. In the minorization stage, we exploit the hidden convexity of the SINR function to derive the minorizing function at a given point. In the maximization stage, we show that the maximization problem enjoys tight convex relaxation and we propose two ways to solve the relaxed maximization problem: one requiring an off-the-shelf solver and the other using the Mirror Descent Algorithm (MDA) [35] framework by introducing an auxiliary simplex. Thus, the proposed algorithms successively solve a series of simple convex problems which enjoy low computational complexity and a fast convergence speed in terms of CPU time.

C. Organization and Notation

The rest of the paper is organized as follows. In Section II, we specify the problem formulation. In Section III, we first give a brief introduction of the vanilla MM method, and then move on to its extension, where the objective takes the pointwise minimum format. In Section IV, we provide the algorithmic framework for solving the maximin problem, i.e., (1). In Section V, we look into specific applications and examples for case studies. Finally, Section VI presents numerical simulations, and the conclusions are given in Section VII.

The following notation is adopted. Boldface upper-case letters represent matrices, boldface lower-case letters denote column vectors, and standard lower-case letters stand for scalars. \mathbb{R} (\mathbb{C}) denotes the real (complex) field. \odot stands for the Hadamard product. $\|\cdot\|_p$ denotes the p -norm of a vector. $\langle \mathbf{x}, \mathbf{y} \rangle$ denotes the inner product of \mathbf{x} and \mathbf{y} . $\nabla(\cdot)$ represents the gradient of a vector (matrix) function (the way to derive the complex-valued gradient follows [36]), and \mathbf{I} stands for the identity matrix. \mathbf{X}^T , \mathbf{X}^* , \mathbf{X}^H , $\text{Tr}(\mathbf{x})$, and $\lambda_{\max}(\mathbf{x})$ denote the transpose, complex conjugate, conjugate transpose, trace, and the largest eigenvalue of \mathbf{X} , respectively. $\text{Diag}(\mathbf{x})$ is a diagonal matrix with \mathbf{x} filling its principal diagonal. $\mathbf{X} \succeq \mathbf{0}$ means \mathbf{X} is positive semidefinite.

II. PROBLEM STATEMENT

We consider the maximization of the minimum of several SINR functions subject to the PAR constraint (cf. (1)) and we specify the SINR functions in this section. We denote the length of the transmitting sequence(s) and receiving filters as N and M , respectively. Recall that \mathbf{s} (or \mathbf{s}_i) is the transmitting sequence and \mathbf{w}_i is the receiving filter. In radar target detection, SINR is expressed as

$$\text{SINR}_i = \frac{\alpha_i |\mathbf{w}_i^H \mathbf{H}_i \mathbf{s}|^2}{\mathbf{w}_i^H \sum_I(\mathbf{s}) \mathbf{w}_i + \mathbf{w}_i^H \mathbf{R} \mathbf{w}_i} \quad (3)$$

¹B-stationarity is recently mentioned in [33] and was proposed in [34].

with

$$\Sigma_I(\mathbf{s}) = \sum_j \beta_j \mathbf{M}_j \mathbf{s} \mathbf{s}^H \mathbf{M}_j^H, \quad (4)$$

while in multiuser communications,

$$\text{SINR}_i = \frac{\alpha_i |\mathbf{w}_i^H \mathbf{H}_i \mathbf{s}_i|^2}{\mathbf{w}_i^H \Sigma_I(\{\mathbf{s}_j\}_{j \neq i}) \mathbf{w}_i + \mathbf{w}_i^H \mathbf{R} \mathbf{w}_i} \quad (5)$$

with

$$\Sigma_I(\{\mathbf{s}_j\}_{j \neq i}) = \sum_{j=1, j \neq i}^I \alpha_j \mathbf{H}_j \mathbf{s}_j \mathbf{s}_j^H \mathbf{H}_j^H. \quad (6)$$

The expression of SINR_i (cf. [10], [13], [16], [21], [37]) is interpreted as follows. The numerator is the power of the desired receiving signal: $\alpha_i > 0$ is the parameter representing the path gain (or loss) and $\mathbf{H}_i \in \mathbb{C}^{M \times N}$ represents the channel matrix. The denominator is the power of the signal-dependent interference (the first term) plus background noise (the second term). The matrices $\Sigma_I(\mathbf{s})$ (cf. [1], [16], [38]) and $\Sigma_I(\{\mathbf{s}_j\}_{j \neq i})$ (cf. [21]) are the interference covariance matrices, with $\beta_j, \alpha_j > 0$. The matrix $\mathbf{M}_j \in \mathbb{C}^{M \times N}$ is an application-dependent constant matrix. The matrix $\mathbf{H}_j \in \mathbb{C}^{M \times N}$ is the channel matrix of the j th user.

III. PRELIMINARIES: THE MM METHOD

A. The Vanilla MM Method

The MM method refers to the Majorization-Minimization method for minimization problems or the Minorization-Maximization method for maximization problems. The MM method [39], [50] can be applied to solve the following general optimization problem:

$$\begin{aligned} & \underset{\mathbf{x}}{\text{maximize}} && F(\mathbf{x}) \\ & \text{subject to} && \mathbf{x} \in \mathcal{X}, \end{aligned} \quad (7)$$

where \mathcal{X} is some constraint set. Rather than maximizing $F(\mathbf{x})$ directly, we consider successively solving a series of simple optimization problems. The algorithm initializes at some feasible starting point $\mathbf{x}^{(0)}$, and then iterates as $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots$ until a convergence criterion is met. The update rule at any iteration, say the n th iteration, is

$$\mathbf{x}^{(n+1)} \in \arg \max_{\mathbf{x} \in \mathcal{X}} \bar{F}(\mathbf{x}, \mathbf{x}^{(n)}), \quad (8)$$

where $\bar{F}(\mathbf{x}, \mathbf{x}^{(n)})$ is a minorizing function of $F(\mathbf{x})$ at $\mathbf{x}^{(n)}$. Suppose \mathcal{X} is a convex set, $\bar{F}(\mathbf{x}, \mathbf{x}^{(n)})$ must satisfy the following conditions so as to claim convergence [40]:

- A1) $\bar{F}(\mathbf{y}, \mathbf{y}) = F(\mathbf{y}), \forall \mathbf{y} \in \mathcal{X}$,
- A2) $\bar{F}(\mathbf{x}, \mathbf{y}) \leq F(\mathbf{x}), \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$,
- A3) $\bar{F}'(\mathbf{y}, \mathbf{y}; \mathbf{d}) = F'(\mathbf{y}; \mathbf{d}), \forall \mathbf{d}$ with $\mathbf{y} + \mathbf{d} \in \mathcal{X}$,
- A4) $\bar{F}(\mathbf{x}, \mathbf{y})$ is continuous in (\mathbf{x}, \mathbf{y}) ,

where F' stands for directional derivative, whose definition is

$$F'(\mathbf{x}; \mathbf{d}) = \lim_{\lambda \downarrow 0} \inf \frac{F(\mathbf{x} + \lambda \mathbf{d}) - F(\mathbf{x})}{\lambda}. \quad (9)$$

The proof of convergence to a stationary point can be found in [40], where it is proved that the limit point $\mathbf{x}^{(\infty)}$ satisfies

$$F'(\mathbf{x}^{(\infty)}; \mathbf{d}) \leq 0, \quad \forall \mathbf{d} \text{ with } \mathbf{x}^{(\infty)} + \mathbf{d} \in \mathcal{X}. \quad (10)$$

But it only applies to the case where \mathcal{X} is a convex set. If \mathcal{X} is nonconvex, we should modify (A3) so as to claim stationarity convergence:

A3) $\bar{F}'(\mathbf{y}, \mathbf{y}; \mathbf{d}) = F'(\mathbf{y}; \mathbf{d}), \forall \mathbf{d} \in \mathcal{T}_{\mathcal{X}}(\mathbf{y})$, where in this case \bar{F} and F are defined on the whole \mathbb{R} or \mathbb{C} space and $\mathcal{T}_{\mathcal{X}}(\mathbf{y})$ is the Bouligand tangent cone of \mathcal{X} at \mathbf{y} . The expression $\mathbf{d} \in \mathcal{T}_{\mathcal{X}}(\mathbf{y})$ means that there exist a sequence of vectors $\{\mathbf{y}^{(k)}\} \subset \mathcal{X}$ converging to \mathbf{y} and a sequence of positive scalars $\{\lambda^{(k)}\}$ converging to 0 such that $\mathbf{d} = \lim_{k \rightarrow \infty} \frac{\mathbf{y}^{(k)} - \mathbf{y}}{\lambda^{(k)}}$. For more details, interested readers may refer to [33], [34]. If (A3) is modified in this way, then we can prove that the limit point $\mathbf{x}^{(\infty)}$ satisfies

$$F'(\mathbf{x}^{(\infty)}; \mathbf{d}) \leq 0, \quad \forall \mathbf{d} \in \mathcal{T}_{\mathcal{X}}(\mathbf{x}^{(\infty)}), \quad (11)$$

and thus B-stationarity is achieved.

B. MM in the Maximin Case

When the objective takes the form of $F(\mathbf{x}) = \min_{i=1, \dots, I} f_i(\mathbf{x})$ (the f_i 's are assumed differentiable), it seems nontrivial to derive a minorizing function satisfying all the aforementioned conditions. The main difficulty is, given that F is nondifferentiable, how to find an \bar{F} that has the same directional derivative as F at a given feasible point. The answer turns out to be simple:

$$\bar{F}(\mathbf{x}, \mathbf{y}) = \min_{i=1, \dots, I} \bar{f}_i(\mathbf{x}, \mathbf{y}), \quad (12)$$

with each \bar{f}_i being a tight lower bound of f_i , satisfying: $\forall i$,

- B1) $\bar{f}_i(\mathbf{y}, \mathbf{y}) = f_i(\mathbf{y}), \forall \mathbf{y} \in \mathcal{X}$,
- B2) $\bar{f}_i(\mathbf{x}, \mathbf{y}) \leq f_i(\mathbf{x}), \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$,
- B3) $\nabla \bar{f}_i(\mathbf{y}, \mathbf{y}) = \nabla f_i(\mathbf{y}), \forall \mathbf{y} \in \mathcal{X}$,
- B4) $\bar{f}_i(\mathbf{x}, \mathbf{y})$ is continuous in (\mathbf{x}, \mathbf{y}) .

To guarantee stationarity convergence, we check whether $\bar{F}(\mathbf{x}, \mathbf{y})$ satisfies (A1)–(A4):

Checking (A1): $\forall \mathbf{y} \in \mathcal{X}$,

$$\bar{F}(\mathbf{y}, \mathbf{y}) = \min_{i=1, \dots, I} \bar{f}_i(\mathbf{y}, \mathbf{y}) = \min_{i=1, \dots, I} f_i(\mathbf{y}) = F(\mathbf{y}). \quad (13)$$

Checking (A2): $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$,

$$\begin{aligned} & \bar{f}_i(\mathbf{x}, \mathbf{y}) \leq f_i(\mathbf{x}) \\ \implies & \min_{i=1, \dots, I} \bar{f}_i(\mathbf{x}, \mathbf{y}) \leq \min_{i=1, \dots, I} f_i(\mathbf{x}) \\ \implies & \bar{F}(\mathbf{x}, \mathbf{y}) \leq F(\mathbf{x}). \end{aligned} \quad (14)$$

Checking (A3): according to [41, Theorem 9.16], given \mathbf{d} , the directional derivative of \bar{F} in (8) can be expressed as

$$\bar{F}'(\mathbf{y}, \mathbf{y}; \mathbf{d}) = \max \{ \langle \xi, \mathbf{d} \rangle : \xi \in \partial \bar{F}(\mathbf{y}, \mathbf{y}) \}, \quad (15)$$

where $\partial \bar{F}(\mathbf{y}, \mathbf{y}) = \text{conv}(\{\nabla \bar{f}_i(\mathbf{y}, \mathbf{y}) : \bar{F}(\mathbf{y}, \mathbf{y}) = \bar{f}_i(\mathbf{y}, \mathbf{y})\})$ and $\text{conv}(\mathcal{A})$ is the convex hull of the set \mathcal{A} . We also derive the directional derivative of F :

$$F'(\mathbf{y}; \mathbf{d}) = \max \{ \langle \xi, \mathbf{d} \rangle : \xi \in \partial F(\mathbf{y}) \}, \quad (16)$$

where $\partial F(\mathbf{y}) = \text{conv}(\{\nabla f_i(\mathbf{y}) : F(\mathbf{y}) = f_i(\mathbf{y})\})$. From $\bar{F}(\mathbf{y}, \mathbf{y}) = F(\mathbf{y})$ and $f_i(\mathbf{y}, \mathbf{y}) = f_i(\mathbf{y})$, we obtain $\{i | \bar{F}(\mathbf{y}, \mathbf{y}) = f_i(\mathbf{y}, \mathbf{y})\} = \{i | F(\mathbf{y}) = f_i(\mathbf{y})\}$. In addition, \bar{f}_i satisfies $\nabla \bar{f}_i(\mathbf{y}, \mathbf{y}) = \nabla f_i(\mathbf{y})$, $\forall i$, so $\partial \bar{F}(\mathbf{y}, \mathbf{y}) = \partial F(\mathbf{y})$. If \mathcal{X} is convex, $\bar{F}'(\mathbf{y}, \mathbf{y}; \mathbf{d}) = F'(\mathbf{y}; \mathbf{d})$, $\forall \mathbf{d}$ with $\mathbf{y} + \mathbf{d} \in \mathcal{X}$. If \mathcal{X} is nonconvex, $\bar{F}'(\mathbf{y}, \mathbf{y}; \mathbf{d}) = F'(\mathbf{y}; \mathbf{d})$, $\forall \mathbf{d} \in \mathcal{T}_{\mathcal{X}}(\mathbf{y})$.

Checking (A4): Obvious.

Thus, the piecewise function $\bar{F}(\mathbf{x}, \mathbf{y})$ in (12) satisfies (A1)–(A4) and the limit point of (8), i.e., $\mathbf{x}^{(\infty)}$, shall satisfy either (10) or (11). With the help of [33], we can further specify what kind of stationary point it is:

- when \mathcal{X} is a convex set, d-stationarity is achieved;
- when \mathcal{X} is a nonconvex set, B-stationarity is achieved.

IV. ALGORITHMIC FRAMEWORK OF MM METHOD

We are now prepared to present the MM algorithmic framework for solving (1). For simplicity, we focus on the case of radar target detection, i.e., (3). The case of multiuser communications follows the same idea despite some minor differences (to be elaborated in the next section). The MM method is naturally split into two stages: the minorization and the maximization.

A. Minorizing Function Construction

Before constructing the minorizing function, we first simplify the original problem by maximizing with respect to \mathbf{w}_i , $\forall i$. It is not hard to show that given \mathbf{s} , the optimal solution for \mathbf{w}_i is (up to a positive scaling factor)

$$\mathbf{w}_i^* = \frac{(\boldsymbol{\Sigma}_I(\mathbf{s}) + \mathbf{R})^{-1} \mathbf{H}_i \mathbf{s}}{\mathbf{s}^H \mathbf{H}_i^H (\boldsymbol{\Sigma}_I(\mathbf{s}) + \mathbf{R})^{-1} \mathbf{H}_i \mathbf{s}}. \quad (17)$$

Then the original problem (1) is reduced to

$$\begin{aligned} & \underset{\mathbf{s} \in \mathbb{C}^N}{\text{maximize}} && \min_{i=1,2,\dots,I} \text{SINR}_i(\mathbf{s}) \\ & \text{subject to} && \mathbf{s} \in \mathcal{S}, \end{aligned} \quad (18)$$

where

$$\text{SINR}_i(\mathbf{s}) = \alpha_i \mathbf{s}^H \mathbf{H}_i^H (\boldsymbol{\Sigma}_I(\mathbf{s}) + \mathbf{R})^{-1} \mathbf{H}_i \mathbf{s}. \quad (19)$$

We already know from Section III-B that finding a minorizing function for $\min_{i=1,2,\dots,I} \text{SINR}_i(\mathbf{s})$ can be boiled down to finding one for each $\text{SINR}_i(\mathbf{s})$. Thus we can focus on the expression of $\text{SINR}_i(\mathbf{s})$ only. In the following, we are going to find a tight lower bound for $\text{SINR}_i(\mathbf{s})$ at the current iteration value $\mathbf{s}^{(n)}$.

We do a change of variable: let \mathbf{G} be $\boldsymbol{\Sigma}_I(\mathbf{s})$, and then $\text{SINR}_i = \text{SINR}_i(\mathbf{s}, \mathbf{G}) = \alpha_i \mathbf{s}^H \mathbf{H}_i^H (\mathbf{G} + \mathbf{R})^{-1} \mathbf{H}_i \mathbf{s}$. We can see that SINR_i is a matrix fractional function and proves to be jointly convex in (\mathbf{s}, \mathbf{G}) , as can be seen from [29, Example 3.4]. A toy example is shown as follows for intuitive illustration.

Example 1: Let \mathbf{s} and \mathbf{G} be scalars, $\mathbf{G} \geq 0$, $\alpha_i = 1$, $\mathbf{H}_i = 2$, $\mathbf{R} = 3$. Thus, $\text{SINR}_i(\mathbf{s}, \mathbf{G}) = \frac{2\mathbf{s} \times 2\mathbf{s}}{\mathbf{G} + 3} = \frac{4\mathbf{s}^2}{\mathbf{G} + 3}$. We can verify the convexity in Figure 1.

In that sense, a simple first-order Taylor expansion with respect to (\mathbf{s}, \mathbf{G}) at $(\mathbf{s}_0, \mathbf{G}_0)$ gives us a tight lower bound:

$$\begin{aligned} \text{SINR}_i(\mathbf{s}, \mathbf{G}) & \geq \text{SINR}_i(\mathbf{s}_0, \mathbf{G}_0) \\ & + 2\text{Re} [\mathbf{b}_i^H (\mathbf{s} - \mathbf{s}_0)] - \text{Tr} (\mathbf{a}_i \mathbf{a}_i^H \cdot (\mathbf{G} - \mathbf{G}_0)), \end{aligned} \quad (20)$$

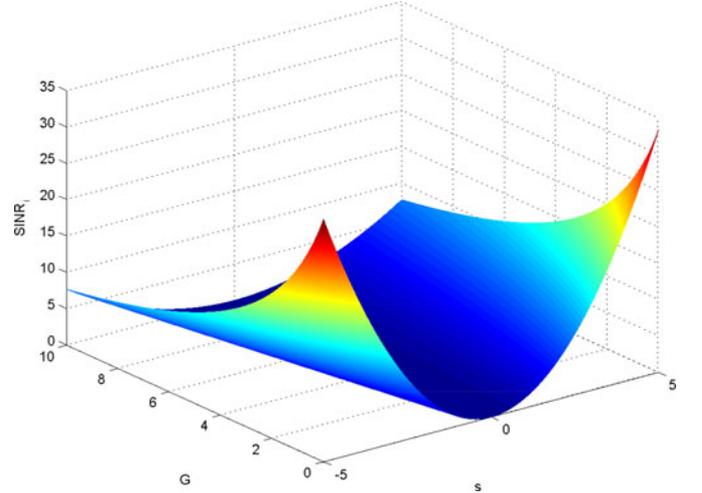


Fig. 1. Convexity of $\text{SINR}_i(\mathbf{s}, \mathbf{G})$.

where

$$\mathbf{b}_i = \alpha_i \mathbf{H}_i^H (\mathbf{G}_0 + \mathbf{R})^{-1} \mathbf{H}_i \mathbf{s}_0, \quad (21)$$

$$\mathbf{a}_i = \sqrt{\alpha_i} (\mathbf{G}_0 + \mathbf{R})^{-1} \mathbf{H}_i \mathbf{s}_0, \quad (22)$$

and $-\mathbf{a}_i \mathbf{a}_i^H$ is the gradient with respect to \mathbf{G} . Now we undo the change of variable $\mathbf{G} = \boldsymbol{\Sigma}_I(\mathbf{s})$ and let \mathbf{s}_0 be $\mathbf{s}^{(n)}$, the transmitting code at the n th iteration:

$$\begin{aligned} \text{SINR}_i(\mathbf{s}, \boldsymbol{\Sigma}_I(\mathbf{s})) & \geq \text{SINR}_i(\mathbf{s}^{(n)}, \boldsymbol{\Sigma}_I(\mathbf{s}^{(n)})) \\ & + 2\text{Re} [\mathbf{b}_i^H (\mathbf{s} - \mathbf{s}^{(n)})] - \text{Tr} (\mathbf{a}_i \mathbf{a}_i^H \cdot (\boldsymbol{\Sigma}_I(\mathbf{s}) - \boldsymbol{\Sigma}_I(\mathbf{s}^{(n)}))), \end{aligned} \quad (23)$$

where \mathbf{b}_i and \mathbf{a}_i should also be adjusted:

$$\mathbf{b}_i = \alpha_i \mathbf{H}_i^H (\boldsymbol{\Sigma}_I(\mathbf{s}^{(n)}) + \mathbf{R})^{-1} \mathbf{H}_i \mathbf{s}^{(n)} \quad (24)$$

and

$$\mathbf{a}_i = \sqrt{\alpha_i} (\boldsymbol{\Sigma}_I(\mathbf{s}^{(n)}) + \mathbf{R})^{-1} \mathbf{H}_i \mathbf{s}^{(n)}. \quad (25)$$

Remark 2: In order to get more intuition of the inequality (23), we continue from Example 1 and set $\mathbf{G} = 5\mathbf{s}^2$ and $\mathbf{s}^{(n)} = 2$. The inequality (23) is illustrated in Figure 2. As we can see in the plot, the blue curve stands for SINR_i and the red curve is a tight lower bound. We are minorizing a nonconvex function with a concave function.

Lemma 3: A minorizing function of $\text{SINR}_i(\mathbf{s})$ at $\mathbf{s} = \mathbf{s}^{(n)}$ is given as

$$\begin{aligned} \overline{\text{SINR}}_i(\mathbf{s}, \mathbf{s}^{(n)}) & = \text{SINR}_i(\mathbf{s}^{(n)}) \\ & + 2\text{Re} [\mathbf{c}_i^H (\mathbf{s} - \mathbf{s}^{(n)})] - \lambda_{u,i} \|\mathbf{s} - \mathbf{s}^{(n)}\|_2^2, \end{aligned} \quad (26)$$

where

$$\mathbf{c}_i = \mathbf{b}_i - \mathbf{A}_i \mathbf{s}^{(n)}, \quad (27)$$

$$\mathbf{A}_i = \sum_j \beta_j \mathbf{M}_j^H \mathbf{a}_i \mathbf{a}_i^H \mathbf{M}_j \succeq \mathbf{0}, \quad (28)$$

and

$$\lambda_{u,i} = \lambda_{\max}(\mathbf{A}_i) > 0. \quad (29)$$

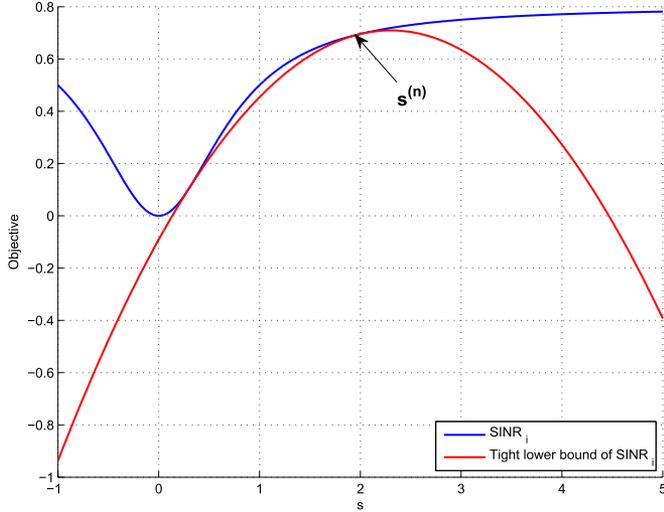


Fig. 2. Construction of a tight lower bound for a nonconvex function.

Proof: See Appendix A for the detailed proof. ■

Then, the minorizing function for $\min_{i=1,2,\dots,I} \text{SINR}_i(\mathbf{s})$ is

$$\min_{i=1,2,\dots,I} \overline{\text{SINR}}_i(\mathbf{s}, \mathbf{s}^{(n)}). \quad (30)$$

According to the framework of the MM method, at every iteration we just need to solve the following problem:

$$\begin{aligned} & \text{maximize} && \min_{i=1,2,\dots,I} \overline{\text{SINR}}_i(\mathbf{s}, \mathbf{s}^{(n)}) \\ & \text{subject to} && \mathbf{s} \in \mathcal{S}. \end{aligned} \quad (31)$$

B. Maximization Solution Pursuit

Since $\|\mathbf{s}\|_2 = 1$ and $\|\mathbf{s}^{(n)}\|_2 = 1$ (from constraint set \mathcal{S}), we can now rewrite (31) as

$$\text{maximize}_{\mathbf{s} \in \mathcal{S}} \min_{i=1,2,\dots,I} d_i + 2\text{Re} \left[\left(\mathbf{c}_i + \lambda_{u,i} \mathbf{s}^{(n)} \right)^H \mathbf{s} \right], \quad (32)$$

where

$$d_i = \text{SINR}_i(\mathbf{s}^{(n)}) - 2\text{Re} \left[\mathbf{c}_i^H \mathbf{s}^{(n)} \right] - 2\lambda_{u,i}. \quad (33)$$

We can rewrite the discrete minimum in (32) as a continuous minimization over a simplex:

$$\begin{aligned} & \min_{i=1,2,\dots,I} d_i + 2\text{Re} \left[\left(\mathbf{c}_i + \lambda_{u,i} \mathbf{s}^{(n)} \right)^H \mathbf{s} \right] \\ & \stackrel{(a)}{=} \min_{\mathbf{p} \in \mathcal{P}} \sum_{i=1}^I p_i \left(d_i + 2\text{Re} \left[\left(\mathbf{c}_i + \lambda_{u,i} \mathbf{s}^{(n)} \right)^H \mathbf{s} \right] \right) \\ & \stackrel{(b)}{=} \min_{\mathbf{p} \in \mathcal{P}} \mathbf{p}^T \mathbf{d} + 2\text{Re} \left[\left(\left(\mathbf{C} + \mathbf{s}^{(n)} \boldsymbol{\lambda}_u^T \right) \mathbf{p} \right)^H \mathbf{s} \right], \end{aligned} \quad (34)$$

where (a) $\mathcal{P} = \{\mathbf{p} \in \mathbb{R}^I | \mathbf{1}^T \mathbf{p} = 1, \mathbf{p} \geq \mathbf{0}\}$ is a simplex and (b) $\mathbf{d} = [d_1, d_2, \dots, d_I]^T$, $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_I]$, and $\boldsymbol{\lambda}_u = [\lambda_{u,1}, \lambda_{u,2}, \dots, \lambda_{u,I}]^T$. Thus, (32) has an equivalent form:

$$\text{max}_{\mathbf{s} \in \mathcal{S}} \min_{\mathbf{p} \in \mathcal{P}} 2\text{Re} \left[\left(\left(\mathbf{C} + \mathbf{s}^{(n)} \boldsymbol{\lambda}_u^T \right) \mathbf{p} \right)^H \mathbf{s} \right] + \mathbf{p}^T \mathbf{d}. \quad (35)$$

Algorithm 1: Accelerated Solver-Based MM Algorithm.

Require: feasible $\mathbf{s}^{(0)}$, $n = 0$;

- 1: **repeat**
- 2: Compute \mathbf{d} , \mathbf{C} , and $\boldsymbol{\lambda}_u$ (cf. (33), (27), and (29), respectively);
- 3: Solve (37) via some off-the-shelf solver and get its optimal solution $\hat{\mathbf{s}}^{(n)}$;
- 4: Apply acceleration technique (39) for step size β ;
- 5: $\mathbf{s}^{(n+1)} = \frac{\mathbf{s}^{(n)} + \beta(\hat{\mathbf{s}}^{(n)} - \mathbf{s}^{(n)})}{\|\mathbf{s}^{(n)} + \beta(\hat{\mathbf{s}}^{(n)} - \mathbf{s}^{(n)})\|_2}$;
- 6: $n = n + 1$;
- 7: **until** convergence

Lemma 4: In problem (35), a saddle point exists and it can be obtained from solving the relaxed problem where the nonconvex constraint set \mathcal{S} (cf. (2)) is relaxed to

$$\mathcal{S}_{\text{relaxed}} = \left\{ \mathbf{s} \in \mathbb{C}^N \mid \|\mathbf{s}\|_2 \leq 1, \|\mathbf{s}\|_\infty \leq \sqrt{\frac{\rho}{N}} \right\}. \quad (36)$$

Proof: See Appendix B for the detailed proof. ■

Now we look into the relaxed problem:

$$\text{maximize}_{\mathbf{s} \in \mathcal{S}_{\text{relaxed}}} \min_{i=1,2,\dots,I} d_i + 2\text{Re} \left[\left(\mathbf{c}_i + \lambda_{u,i} \mathbf{s}^{(l)} \right)^H \mathbf{s} \right], \quad (37)$$

or equivalently

$$\text{max}_{\mathbf{s} \in \mathcal{S}_{\text{relaxed}}} \min_{\mathbf{p} \in \mathcal{P}} 2\text{Re} \left[\left(\left(\mathbf{C} + \mathbf{s}^{(l)} \boldsymbol{\lambda}_u^T \right) \mathbf{p} \right)^H \mathbf{s} \right] + \mathbf{p}^T \mathbf{d}. \quad (38)$$

Problem (38) is derived from (35) by changing \mathcal{S} to $\mathcal{S}_{\text{relaxed}}$ and problem (37) is a reformulation of (38) into the discrete minimum format. We are going to propose two approaches for solving the relaxed problem.

The First Approach: If we focus on (37), we can solve it via an off-the-shelf solver directly. To accelerate the convergence speed of the MM algorithm, we adopt the following technique:² at any iteration, say the n th iteration, we utilize the optimal \mathbf{s} , denoted by $\hat{\mathbf{s}}^{(n)}$, to provide an ascent direction, $\hat{\mathbf{s}}^{(n)} - \mathbf{s}^{(n)}$, and do the line search as [42]:

choose $\alpha (> 1)$;
 $\beta = 1$;

$$\mathbf{s}_{\text{temp}} = \frac{\mathbf{s}^{(n)} + \alpha\beta(\hat{\mathbf{s}}^{(n)} - \mathbf{s}^{(n)})}{\|\mathbf{s}^{(n)} + \alpha\beta(\hat{\mathbf{s}}^{(n)} - \mathbf{s}^{(n)})\|_2};$$

while $\min_{i=1,2,\dots,I} \text{SINR}_i(\mathbf{s}_{\text{temp}}) \geq \min_{i=1,2,\dots,I} \text{SINR}_i(\hat{\mathbf{s}}^{(n)})$

and $\|\mathbf{s}_{\text{temp}}\|_\infty \leq \sqrt{\frac{\rho}{N}}$
 $\beta = \alpha\beta$;

$$\mathbf{s}_{\text{temp}} = \frac{\mathbf{s}^{(n)} + \alpha\beta(\hat{\mathbf{s}}^{(n)} - \mathbf{s}^{(n)})}{\|\mathbf{s}^{(n)} + \alpha\beta(\hat{\mathbf{s}}^{(n)} - \mathbf{s}^{(n)})\|_2};$$

end

(39)

The first algorithm is summarized in Algorithm 1.

²The convergence result also holds for the accelerated MM algorithm and the proof mostly follows that of [40, Theorem 1] with slight modifications on one equation: (following the notations and problem settings of [40]) $u(x, x^{rj}) \geq u(\text{MM}(x^{rj}), x^{rj}) \geq f(\text{MM}(x^{rj})) \geq f(x^{rj+1}) \geq f(x^{rj+1}) = u(x^{rj+1}, x^{rj+1})$, where $\text{MM}(\cdot)$ is the MM algorithm mapping and x^{rj+1} is the next iteration point found by the acceleration technique. In this case, subsequence stationarity convergence is maintained.

The Second Approach: Now we focus on (38). The objective function in (38) is bilinear in \mathbf{s} and \mathbf{p} ; $\mathcal{S}_{\text{relaxed}}$ and \mathcal{P} are both nonempty compact convex sets. Following the results of [43, Corollary 37.6.2] and [43, Lemma 36.2], a saddle point exists and we can swap maximin to be minimax without affecting the solutions:

$$\min_{\mathbf{p} \in \mathcal{P}} \max_{\mathbf{s} \in \mathcal{S}_{\text{relaxed}}} 2\text{Re} \left[\left((\mathbf{C} + \mathbf{s}^{(n)} \lambda_u^T) \mathbf{p} \right)^H \mathbf{s} \right] + \mathbf{p}^T \mathbf{d}, \quad (40)$$

which can be compactly rewritten as

$$\text{minimize}_{\mathbf{p} \in \mathcal{P}} h(\mathbf{p}), \quad (41)$$

where

$$h(\mathbf{p}) = \max_{\mathbf{s} \in \mathcal{S}_{\text{relaxed}}} 2\text{Re} \left[(\mathbf{B}\mathbf{p})^H \mathbf{s} \right] + \mathbf{p}^T \mathbf{d} \quad (42)$$

and $\mathbf{B} = \mathbf{C} + \mathbf{s}^{(n)} \lambda_u^T$. In particular,

- when $\rho = 1$, $\mathcal{S}_{\text{relaxed}} = \left\{ \mathbf{s} \in \mathbb{C}^N \mid \|\mathbf{s}\|_\infty \leq \sqrt{\frac{1}{N}} \right\}$ and $h(\mathbf{p}) = 2\sqrt{\frac{1}{N}} \|\mathbf{B}\mathbf{p}\|_1 + \mathbf{p}^T \mathbf{d}$;
- when $\rho = N$, $\mathcal{S}_{\text{relaxed}} = \left\{ \mathbf{s} \in \mathbb{C}^N \mid \|\mathbf{s}\|_2 \leq 1 \right\}$ and $h(\mathbf{p}) = 2\|\mathbf{B}\mathbf{p}\|_2 + \mathbf{p}^T \mathbf{d}$.

We solve (41) via the MDA, which iteratively repeats the following three steps until convergence:

- 1) Get subgradient $\mathbf{h}^{(m)} \in \partial h(\mathbf{p}^{(m)})$;
- 2) Update $\mathbf{p}^{(m+1)} = \arg \min_{\mathbf{p} \in \mathcal{P}} \left\{ \mathbf{h}^{(m)T} \mathbf{p} + \frac{1}{\gamma_m} B_\Psi(\mathbf{p}, \mathbf{p}^{(m)}) \right\}$, where $B_\Psi(\mathbf{p}, \mathbf{p}^{(m)}) = \Psi(\mathbf{p}) - \Psi(\mathbf{p}^{(m)}) - \nabla^T \Psi(\mathbf{p}^{(m)}) (\mathbf{p} - \mathbf{p}^{(m)})$;
- 3) $m = m + 1$, and go to step 1) unless convergence is achieved.

Following [35], when \mathcal{P} is the unit simplex, one can choose

$$\Psi(\mathbf{p}) = \begin{cases} \sum_{i=1}^I p_i \log p_i & \mathbf{p} \in \mathcal{P} \\ +\infty & \text{otherwise} \end{cases}, \quad (43)$$

and the update step 2) is simplified to (“exp” operation is imposed in an elementwise way)

$$\mathbf{p}^{(m+1)} = \frac{\mathbf{p}^{(m)} \odot \exp(-\gamma_m \mathbf{h}^{(m)})}{\mathbf{1}^T (\mathbf{p}^{(m)} \odot \exp(-\gamma_m \mathbf{h}^{(m)}))}. \quad (44)$$

The choice of $\{\gamma_m\}$ also follows [35]:

$$\gamma_m = \frac{\mathcal{O}(1)}{\sqrt{m}}, \quad (45)$$

where $\mathcal{O}(1)$ represents some constant. The MDA algorithm is summarized in Algorithm 2. MDA is terminated when the improvement between iterations is smaller than a threshold (by default 10^{-5}) or the number of iterations reaches a predetermined maximum (by default 3000).

Now we are only left with computing $\mathbf{h}^{(m)}$, the update step 1). The subgradient $\mathbf{h}^{(m)}$ is given as

$$\mathbf{h}^{(m)} = 2\text{Re} \left[\mathbf{B}^H \mathbf{x}^{(m)} \right] + \mathbf{d}, \quad (46)$$

Algorithm 2: MDA Algorithm.

- Require:** feasible $\mathbf{p}^{(0)}$, $m = 0$;
- 1: **repeat**
 - 2: Get subgradient: $\mathbf{h}^{(m)} \in \partial h(\mathbf{p}^{(m)})$;
 - 3: $\mathbf{p}^{(m+1)} = \frac{\mathbf{p}^{(m)} \odot \exp(-\gamma_m \mathbf{h}^{(m)})}{\mathbf{1}^T (\mathbf{p}^{(m)} \odot \exp(-\gamma_m \mathbf{h}^{(m)}))}$;
 - 4: $m = m + 1$;
 - 5: **until** convergence
-

Algorithm 3: Accelerated MDA-Based MM Algorithm.

- Require:** feasible $\mathbf{s}^{(0)}$, $n = 0$;
- 1: **repeat**
 - 2: Compute \mathbf{d} , \mathbf{C} , and λ_u (cf. (33), (27), and (29), respectively);
 - 3: Solve (41) via MDA (Algorithm 2) for the optimal \mathbf{p} , denoted as \mathbf{p}^* , and $\hat{\mathbf{s}}^{(n)}$;
 - 4: Apply acceleration technique (39) for step size β ;
 - 5: $\mathbf{s}^{(n+1)} = \frac{\mathbf{s}^{(n)} + \beta(\hat{\mathbf{s}}^{(n)} - \mathbf{s}^{(n)})}{\|\mathbf{s}^{(n)} + \beta(\hat{\mathbf{s}}^{(n)} - \mathbf{s}^{(n)})\|_2}$;
 - 6: $n = n + 1$;
 - 7: **until** convergence
-

where $\mathbf{x}^{(m)} = \arg \max_{\mathbf{x} \in \mathcal{S}_{\text{relaxed}}} \text{Re}[(\mathbf{B}\mathbf{p}^{(m)})^H \mathbf{x}]$. In particular,

- when $\rho = 1$, $\mathbf{x}^{(m)} = \sqrt{\frac{1}{N}} \left(|\mathbf{B}\mathbf{p}^{(m)}|^{-1} \odot [\mathbf{B}\mathbf{p}^{(m)}] \right)$ ($|\cdot|^{-1}$ operation is imposed elementwisely);
- when $\rho = N$, $\mathbf{x}^{(m)} = (\mathbf{B}\mathbf{p}^{(m)}) / \|\mathbf{B}\mathbf{p}^{(m)}\|_2$;
- when $1 < \rho < N$, $\mathbf{x}^{(m)}$ follows the closed-form solution in [2, Algorithm 2]. The phases of $\mathbf{x}^{(m)}$ are aligned with those of $\mathbf{B}\mathbf{p}^{(m)}$. Denote the number of nonzero elements of $\mathbf{B}\mathbf{p}^{(m)}$ as K ($\leq N$), and the set containing all the corresponding indexes as \mathcal{K} . The solution of $|\mathbf{x}^{(m)}|$ is as follows:
 - if $K\rho/N \leq 1$, the solution is

$$|\mathbf{x}_n^{(m)}| = \begin{cases} \sqrt{\frac{\rho}{N}} & \forall n \in \mathcal{K}, \\ \sqrt{\frac{1-K\rho/N}{N-K}} & \forall n \notin \mathcal{K}; \end{cases} \quad (47)$$

– if $K\rho/N > 1$, the solution is

$$|\mathbf{x}^{(m)}| = \left[\eta |\mathbf{B}\mathbf{p}^{(m)}| \right]_0^{\sqrt{\rho/N}}, \quad (48)$$

where η satisfies $\|[\eta |\mathbf{B}\mathbf{p}^{(m)}|]_0^{\sqrt{\rho/N}}\|_2 = 1$ ($|\cdot|$ denotes the elementwise absolute value and $[\mathbf{x}]_a^b$ means projecting \mathbf{x} elementwisely onto $[a, b]$). Observing that $g(\eta) = \|[\eta |\mathbf{B}\mathbf{p}^{(m)}|]_0^{\sqrt{\rho/N}}\|_2$ is a strictly increasing function on

$$\left[0, \frac{\sqrt{\rho/N}}{\min_{n \in \mathcal{K}} \{ |(\mathbf{B}\mathbf{p}^{(m)})_n| \}} \right],$$

there is a unique η satisfying $g(\eta) = 1$. The second algorithm is finally summarized in Algorithm 3.

C. Computational Complexity

Now we discuss the computational complexity of Algorithm 1 and 3. The only difference between the two algorithms is the way they solve the subproblem (31). We analyze the computational complexity on a per-iteration basis or, more precisely, on a per-outer-iteration basis. For analytical convenience, we focus on the deterministic cost only. The deterministic computational cost mainly comes from two sources: 1) computing \mathbf{d} , \mathbf{C} , and λ_u , and 2) solving the simple convex problem (37) or (38). We assume M and N are of the same order ($\mathbf{H}_i, \mathbf{M}_j \in \mathbb{C}^{M \times N}$).

First we look into the computation of \mathbf{d} , \mathbf{C} , and λ_u (cf. (33), (27), and (29), respectively). The most costly operation in computing one element of $\mathbf{d} \in \mathbb{R}^I$ and one column of $\mathbf{C} \in \mathbb{C}^{N \times I}$ needs $\mathcal{O}(N^3)$ because of $(\Sigma_I(\mathbf{s}^{(n)}) + \mathbf{R})^{-1}$, so the overall complexity is $\mathcal{O}(IN^3)$. Recall that $\lambda_{u,i} = \lambda_{\max}(\mathbf{A}_i)$, where $\mathbf{A}_i \succeq \mathbf{0}$. The computation of $\lambda_{u,i}$ can be replaced by $\text{Tr}(\mathbf{A}_i)$ in practice because, first, this change does not violate any of the inequalities in the algorithm design, and second, computing $\text{Tr}(\mathbf{A}_i)$ is very efficient, only $\mathcal{O}(N)$. So the overall cost is $\mathcal{O}(IN)$. To this moment, the first source contributes a total amount of complexity $\mathcal{O}(IN^3)$, neglecting the lower-order terms.

Next we move on to the simple convex problem. An off-the-shelf solver, e.g., MOSEK [44], will reformulate the problem into the epigraph form with one more slack variable. Thus, we have I linear constraints. The ℓ_2 - and ℓ_∞ -norm constraints can be rewritten as Second-Order Cone (SOC) constraints: 1) ℓ_2 : $\|\mathbf{s}\|_2 \leq 1$ and 2) ℓ_∞ : $\forall n, |s_n| \leq \sqrt{\frac{P}{N}} \implies \|\text{Re}[s_n], \text{Im}[s_n]\|_2 \leq \sqrt{\frac{P}{N}}$, hence a total of $N + 1$ SOC constraints. To sum up, there are I linear constraints and $N + 1$ SOC constraints, so the computational complexity of solving the simple convex problem should be upper bounded by $\mathcal{O}(N^{3.5})$, the same order as SOCP. If we take a closer look at those SOC constraints, we find that they are of very small size (only two variables) and much simpler than those in [13]: no Hadamard product, no matrix decomposition, and no affine transformation. The resulting SOCP is quite sparse, and modern conic solvers such as MOSEK can exploit the sparsity of the problem very efficiently. That's why the practical complexity is far below the worst-case complexity $\mathcal{O}(N^{3.5})$.

When we solve the convex problem with MDA, the analysis on its computational cost is elaborated in [35, Theorem 5.1], which indicates the gap between the global optimal objective and the best objective achieved in k iterations is no more than $\frac{\mathcal{O}(1)\sqrt{\log I}}{\sqrt{k}}$. The per iteration complexity of MDA is elaborated as follows. MDA consists of two main steps in each iteration: 1) computation of subgradient $\mathbf{h}^{(m)}$: this step involves matrix multiplications $\mathbf{B}\mathbf{p}^{(m)}$ and $\mathbf{B}^H \mathbf{x}^{(m)}$, of complexity $\mathcal{O}(NI)$ ($\mathbf{B} \in \mathbb{C}^{N \times I}$, $\mathbf{p}^{(m)} \in \mathbb{R}^I$, and $\mathbf{x}^{(m)} \in \mathbb{C}^N$); 2) update of $\mathbf{p}^{(m)}$ to $\mathbf{p}^{(m+1)}$: this step involves elementwise exponent, Hadamard product, and summation, of complexity $\mathcal{O}(I)$. Therefore, the per iteration complexity of MDA is $\mathcal{O}(NI)$, neglecting lower order terms.

V. APPLICATIONS AND EXAMPLES

In this section, we specify the expressions for the channel matrix \mathbf{H}_i , the interference covariance matrix $\Sigma_I(\mathbf{s})$ and

$\Sigma_I(\{\mathbf{s}_j\}_{j \neq i})$ etc. in various radar and communications applications. The numerical simulations in the next section will be based on these applications.

A. Radar Application

In a real-life radar system, the target information may not be precisely known, but is believed to lie in a small interval centering some nominal value. In the following, we look into two examples, one in Doppler robust design and the other in a colocated MIMO radar system.

1) *Doppler Robust Design - Monostatic Radar System Transmitting Slow-Time Codes*: Following the setting in [13], we set the transmitting sequence length to be N and $M = N$ (the filter has the same length as the sequence); the channel matrix is given as

$$\mathbf{H}_i = \text{Diag}(\mathbf{p}(\nu_{d_T}^i)), \quad (49)$$

where $\mathbf{p}(\nu) = [1, e^{j2\pi\nu}, \dots, e^{j2\pi(N-1)\nu}]^T$ is the Doppler steering vector and $\nu_{d_T}^i$ is the i th sampled normalized target Doppler frequency, falling within $[\nu_{d_T, \text{lower}}, \nu_{d_T, \text{upper}}]$. The interference covariance matrix $\Sigma_I(\mathbf{s})$ is specifically expressed as

$$\Sigma_I(\mathbf{s}) = \sum_{n_c=0}^{N_c-1} \sum_{l=0}^{L-1} \sigma_{(n_c,l)}^2 \mathbf{J}_{n_c} \left(\Phi_{\epsilon_{(n_c,l)}}^{\bar{\nu}_{d_{(n_c,l)}}} \odot \mathbf{s}\mathbf{s}^H \right) \mathbf{J}_{n_c}^H, \quad (50)$$

where $N_c (< N)$ is the number of range rings, L is the number of azimuth sectors, the range-azimuth bin is denoted as (n_c, l) , the bin of interest is $(0, 0)$ (where we receive signals), $\sigma_{(n_c,l)}^2$ is the mean interfering power associated with the clutter patch located at the range-azimuth bin (n_c, l) whose (normalized) Doppler shift $\nu_{d_{(n_c,l)}}$ is uniformly distributed over the interval $(\bar{\nu}_{d_{(n_c,l)}} - \epsilon_{(n_c,l)}/2, \bar{\nu}_{d_{(n_c,l)}} + \epsilon_{(n_c,l)}/2)$, \mathbf{J}_{n_c} is a Toeplitz matrix with the n_c th subdiagonal entries being 1 and 0 elsewhere, and $\Phi_{\epsilon_{(n_c,l)}}^{\bar{\nu}_{d_{(n_c,l)}}}$ is the covariance matrix of $\mathbf{p}(\nu_{d_{(n_c,l)}})$, given as

$$\Phi_{\epsilon_{(n_c,l)}}^{\bar{\nu}_{d_{(n_c,l)}}}(m, n) = \exp(j2\pi\bar{\nu}_{d_{(n_c,l)}}(m-n)) \times \text{sinc}(\epsilon_{(n_c,l)}(m-n)), \quad (51)$$

$\text{sinc}(x) = \sin(\pi x) / (\pi x)$. In this case, the expression of \mathbf{A}_i is specified as

$$\mathbf{A}_i = \sum_{r=0}^{N_c-1} \sum_{k=0}^{L-1} \mathbf{A}_{i,(r,k)} \odot \left(\Phi_{\epsilon_{(r,k)}}^{\bar{\nu}_{d_{(r,k)}}} \right)^T, \quad (52)$$

where

$$\mathbf{A}_{i,(n_c,l)} = -\sigma_{(n_c,l)}^2 \mathbf{J}_{n_c}^H \mathbf{a}_i \mathbf{a}_i^H \mathbf{J}_{n_c} \quad (53)$$

(\mathbf{a}_i cf. (25)).

2) *Colocated MIMO Radar System*: Following the settings in [16], we set $M = N_s N_r$ and $N = N_s N_t$, where N_s is the number of samples, N_t is the number of transmitting antennas, and N_r is the number of receiving antennas. In this case M and N may be unequal. The number of interference sources is denoted as J ; θ_0 is the angle of the target and θ_j is the angle of

the j th interference source ($j = 1, 2, \dots, J$). We also define

$$\mathbf{F}(\theta) = \mathbf{I}_{N_s} \otimes [\mathbf{f}_r(\theta) \mathbf{f}_t^T(\theta)], \quad (54)$$

$$\mathbf{f}_r(\theta) = \frac{1}{\sqrt{N_r}} [1, e^{-j\pi \sin \theta}, \dots, e^{-j\pi(N_r-1) \sin \theta}]^T, \quad (55)$$

$$\mathbf{f}_t(\theta) = \frac{1}{\sqrt{N_t}} [1, e^{-j\pi \sin \theta}, \dots, e^{-j\pi(N_t-1) \sin \theta}]^T, \quad (56)$$

and the following short notations are adopted: $\mathbf{F}_j = \mathbf{F}(\theta_j), \forall j$ and $\mathbf{F}_{0,i} = \mathbf{F}(\theta_{0,i})$, where $\theta_{0,i}$ is the i th sampled target location angle, falling within $[\theta_{0,\text{lower}}, \theta_{0,\text{upper}}]$. Thus, we specify the channel matrix and the interference covariance matrix:

$$\mathbf{H}_i = \mathbf{F}_{0,i} \quad (57)$$

and

$$\Sigma_I(\mathbf{s}) = \sum_{j=1}^J \beta_j \mathbf{F}_j \mathbf{s} \mathbf{s}^H \mathbf{F}_j^H, \quad (58)$$

where β_j is some positive scaling factor. In this case, the expression of \mathbf{A}_i follows (28) by replacing \mathbf{M}_j with \mathbf{F}_j .

B. Synchronous DS-CDMA Application

Following [21], we consider a reverse-link synchronous DS-CDMA system with I users. The transmitting sequence length is N and $M = N$ (the filter has the same length as the sequence). The channel matrix is given as

$$\mathbf{H}_i = \sum_{l=1}^L h_{i,l} \mathbf{J}_{l-1}, \quad (59)$$

$h_{i,l}$ is the l th fading gains for user i , \mathbf{J}_l is a Toeplitz matrix with the l th subdiagonal entries being 1 and 0 elsewhere, $\mathbf{J}_0 = \mathbf{I}$, and L is the number of fading paths; $\Sigma_I(\{\mathbf{s}_j\}_{j \neq i})$ (cf. (6)) is the covariance matrix measuring the interference to the i th user from the other $I - 1$ users.

We can still use the aforementioned MM algorithms, but some adjustments have to be made. To avoid unnecessary repetition, we give directly the following lemma for $\overline{\text{SINR}}_i$.

Lemma 5: In the multiuser communications example, a minorizing function of $\text{SINR}_i(\{\mathbf{s}_i\})$ at $\{\mathbf{s}_i = \mathbf{s}_i^{(n)}\}$ is given as

$$\begin{aligned} & \overline{\text{SINR}}_i(\{\mathbf{s}_i\}, \{\mathbf{s}_i^{(n)}\}) \\ & \triangleq \text{SINR}_i(\{\mathbf{s}_i^{(n)}\}) + 2\text{Re}[\mathbf{c}_{i,i}^H(\mathbf{s}_i - \mathbf{s}_i^{(l)})] \\ & + \sum_{j=1, j \neq i}^I \left(2\text{Re}[\mathbf{c}_{i,j}^H(\mathbf{s}_j - \mathbf{s}_j^{(l)})] - \lambda_{u,i,j} \|\mathbf{s}_j - \mathbf{s}_j^{(l)}\|_2^2 \right), \end{aligned} \quad (60)$$

where $\forall i$,

$$\mathbf{c}_{i,i} = \alpha_i \mathbf{H}_i^H \left(\Sigma_I(\{\mathbf{s}_j^{(n)}\}_{j \neq i}) + \sigma_n^2 \mathbf{I} \right)^{-1} \mathbf{H}_i \mathbf{s}_i^{(n)}, \quad (61)$$

$\forall j \neq i$,

$$\mathbf{c}_{i,j} = -\mathbf{A}_{i,j}^H \mathbf{s}_j^{(n)}, \quad (62)$$

$$\mathbf{A}_{i,j} = \alpha_j \mathbf{H}_j^H \mathbf{a}_i \mathbf{a}_i^H \mathbf{H}_j \succeq \mathbf{0}, \quad (63)$$

$$\lambda_{u,i,j} = \lambda_{\max}(\mathbf{A}_{i,j}) = \alpha_j \mathbf{a}_i^H \mathbf{H}_j \mathbf{H}_j^H \mathbf{a}_i > 0, \quad (64)$$

and

$$\mathbf{a}_i = \sqrt{\alpha_i} \left(\Sigma_I(\{\mathbf{s}_j^{(n)}\}_{j \neq i}) + \sigma_n^2 \mathbf{I} \right)^{-1} \mathbf{H}_i \mathbf{s}_i^{(n)}. \quad (65)$$

Proof: The proof follows that of Lemma 3 and is thus omitted. \blacksquare

The relaxation argument for the constraint set \mathcal{S} stills holds, which follows that of Lemma 4, and the relaxed problem reads

$$\begin{aligned} & \underset{\mathbf{s}_i \in \mathcal{S}_{\text{relaxed}}, \forall i=1,2,\dots,I}{\text{maximize}} \quad \min_{i=1,2,\dots,I} d_i + 2\text{Re}[\mathbf{c}_{i,i}^H \mathbf{s}_i] \\ & + \sum_{j=1, j \neq i}^I 2\text{Re} \left[\left(\mathbf{c}_{i,j} + \lambda_{u,i,j} \mathbf{s}_j^{(l)} \right)^H \mathbf{s}_j \right], \end{aligned} \quad (66)$$

where $d_i = f_i(\{\mathbf{s}_i^{(l)}\}) - 2\text{Re}[\mathbf{c}_{i,i}^H \mathbf{s}_i^{(l)}] - \sum_{j=1, j \neq i}^I (2\text{Re}[\mathbf{c}_{i,j}^H \mathbf{s}_j^{(l)}] + 2\lambda_{u,i,j})$, or equivalently, by introducing an auxiliary simplex,

$$\begin{aligned} & \max_{\mathbf{s}_i \in \mathcal{S}_{\text{relaxed}}, \forall i} \quad \min_{\mathbf{p} \in \mathcal{P}} 2 \sum_{i=1}^I \text{Re} \left[\left((\mathbf{C}_i + \mathbf{s}_i^{(l)} \boldsymbol{\lambda}_{u,i}^T) \mathbf{p} \right)^H \mathbf{s}_i \right] + \mathbf{p}^T \mathbf{d}, \end{aligned} \quad (67)$$

where $\mathcal{P} = \{\mathbf{p} \in \mathbb{R}^I | \mathbf{1}^T \mathbf{p} = 1, \mathbf{p} \geq \mathbf{0}\}$, $\mathbf{d} = [d_1, d_2, \dots, d_I]^T, \forall i$, $\mathbf{C}_i = [\mathbf{c}_{1,i}, \mathbf{c}_{2,i}, \dots, \mathbf{c}_{I,i}]$, $\boldsymbol{\lambda}_{u,i} = [\lambda_{u,1,i}, \lambda_{u,2,i}, \dots, \lambda_{u,I,i}]^T$, and $\lambda_{u,i,i} = 0$. Then we can use an off-the-shelf solver (following the first approach) or MDA (following the second approach) to solve the maximization problem.

VI. NUMERICAL SIMULATIONS

We present numerical results with respect to two applications, the Doppler robust design and the synchronous DS-CDMA. To avoid redundancy, we omit the colocated MIMO radar example, which is merely a change of constant parameters compared with the Doppler robust design example. All experiments were performed on a PC with a 3.20 GHz i5-4570 CPU and 8 GB RAM. The off-the-shelf solver is specified as 1) MOSEK [44] built in the CVX toolbox [45], shortly denoted as CVX, and/or 2) the Fusion Matlab API in MOSEK, shortly denoted as MOSEK. We include the in-built Matlab nonconvex optimization solver, namely, `fmincon`, as a potential benchmark. The proposed algorithms are terminated when the improvement between iterations is smaller than a threshold (by default 10^{-6}) or the number of iterations reaches a predetermined maximum (by default 500).

A. Doppler Robust Design

Experiment Settings: The transmitting sequence length is $N = 20$ by default. We assume $N_c = 2$ interfering range rings and $L = 100$ azimuth sectors. A homogeneous ground clutter is adopted: $\forall (n_c, l)$, a uniformly distributed clutter is assumed with $\sigma_{(n_c, l)}^2 = \sigma^2 = 1000$ and the Doppler shift of the clutter scatterer $\nu_{d(n_c, l)}$ is uniformly distributed over $\Omega_c = (\bar{\nu}_{d(n_c, l)} - \epsilon_{(n_c, l)}/2, \bar{\nu}_{d(n_c, l)} + \epsilon_{(n_c, l)}/2) = (-0.065, 0.065)$. As for the target, $\alpha_i = \alpha = 10$ dB, $\forall i$. The background noise covariance matrix \mathbf{R} is \mathbf{I} (white noise) by default. The filter bank is designed by assuming $\nu_{d_T}^i \notin \Omega_c, \forall i$, i.e., the uncertainty interval of the target Doppler frequency $\Omega_T = [\nu_{d_T, \text{lower}}, \nu_{d_T, \text{upper}}]$ does not overlap with Ω_c . We set $\Omega_T = [0.34, 0.5]$. The number of filters is $I = 10$ by default.

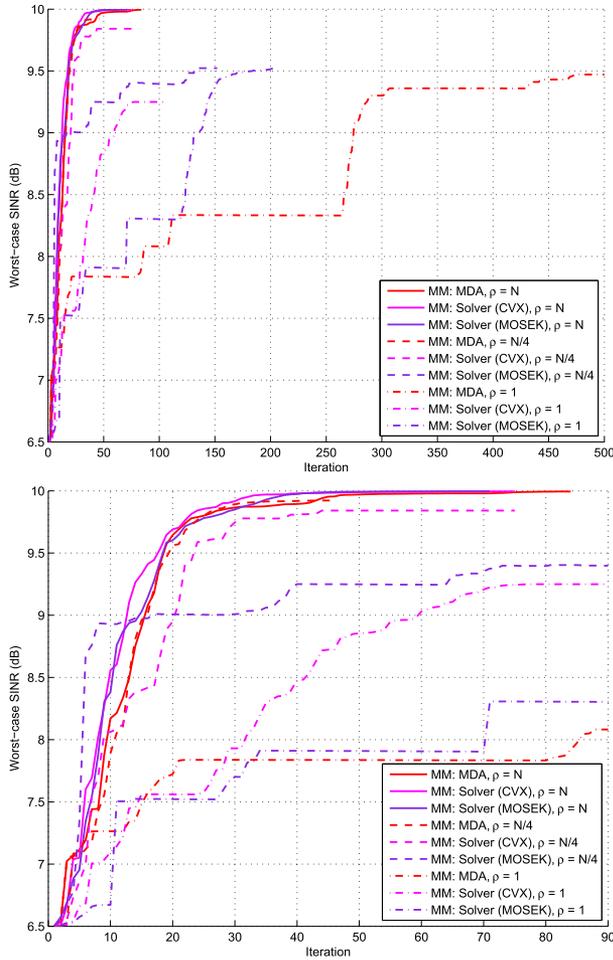


Fig. 3. Convergence plot: worst-case SINR versus iteration, $N = 20$. The lower plot is zoomed in from the upper plot within the iteration interval $[0, 90]$.

For the PAR constraint threshold, we set $\rho = 1$, $N/4$, and N for performance comparison.

1) *Monotonic Property of the Proposed Algorithms:* We implement both Solver-based MM and MDA-based MM; both algorithms initiate from a known sequence: the generalized Barker code (with unit energy). In Figures 3 and 4, we show the monotonic property of the proposed algorithms. The worst-case SINR (i.e., the objective function value) monotonically increases with the number of iterations as well as the time, until it becomes saturated at a certain level. When we increase the parameter ρ , the optimized worst-case SINR also increases because the constraint set becomes more and more relaxed. We may notice that the two algorithms need different numbers of iterations and time to converge, and they may not converge to exactly the same solution. In the current settings, when $\rho = N/4$ and N , MDA-based MM reaches a slightly higher optimized value, while Solver-based MM converges slightly faster, especially in the case of MOSEK; when $\rho = 1$, Solver-based MM using MOSEK directly reaches the highest optimized value and converges the fastest.

It is also nice to see how $\{\text{SINR}_i\}$ evolve for various i through the iterations. We set $\rho = N$. In Figure 5, the evolution process is shown. We can see that any single SINR_i has a general trend of

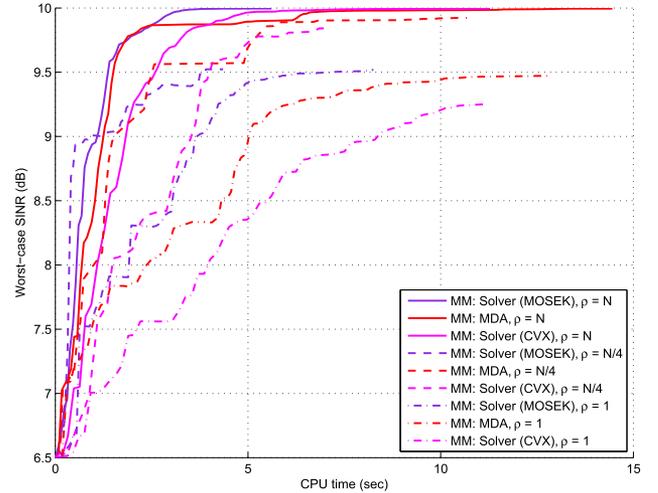


Fig. 4. Convergence plot: worst-case SINR versus CPU time, $N = 20$.

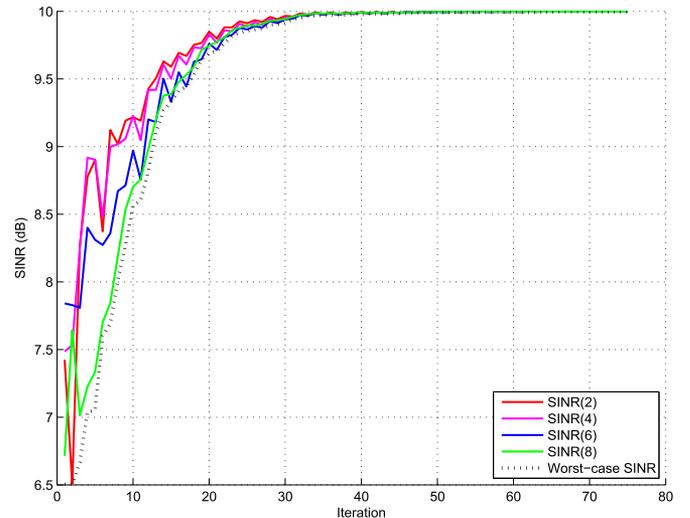


Fig. 5. Convergence plot: $\{\text{SINR}_i\}$ versus iteration for various i , $I = 10$. Please note that only $i = 2, 4, 6, 8$ are displayed.

increase but the evolution process is not monotonic and displays oscillation instead. The worst-case SINR is the minimum of $\{\text{SINR}_i\}$, and its evolution enjoys monotonicity.

2) *Robust Versus Non-robust Design:* We adopt the MDA-based MM as the proposed method, which initiates from the generalized Barker code (with unit energy). In the non-robust design, only the nominal target Doppler frequency is considered. Here we set the nominal value to be the center of the uncertainty interval, i.e., $\hat{\nu}_{dT} = (0.34 + 0.5) / 2 = 0.42$. Some previous works, like [38], have mentioned this non-robust design. We find that our proposed algorithm can serve the same purpose by setting $\Omega_T = \{0.42\}$ and $I = 1$, which turns out to be more efficient since [38] involves SDP in the algorithmic design. The radar detection performance is measured by $\text{SINR}(\nu)$, which is defined as

$$\text{SINR}(\nu) = \max_{i=1,2,\dots,I} \frac{\alpha_i |\mathbf{w}_i^H \mathbf{H}(\nu) \mathbf{s}|^2}{\mathbf{w}_i^H \boldsymbol{\Sigma}_I(\mathbf{s}) \mathbf{w}_i + \mathbf{w}_i^H \mathbf{R} \mathbf{w}_i}, \quad (68)$$

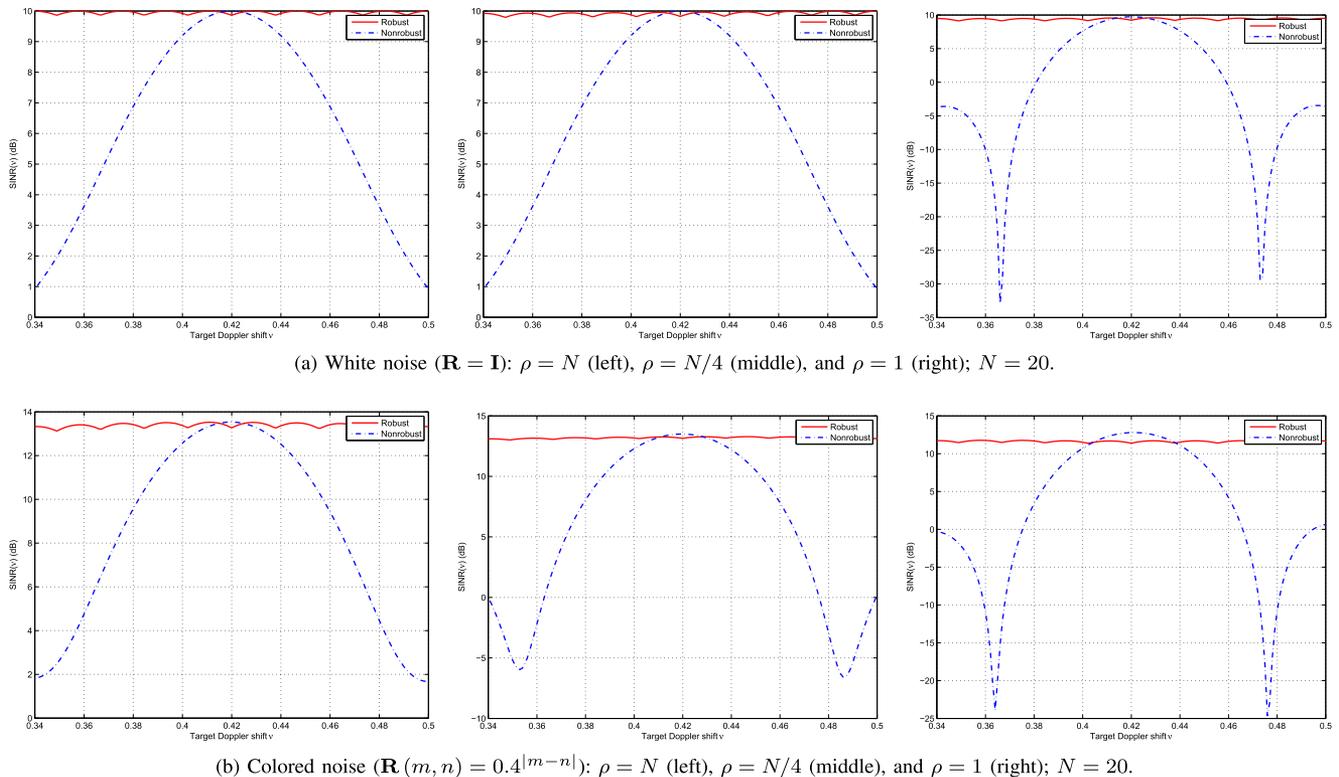


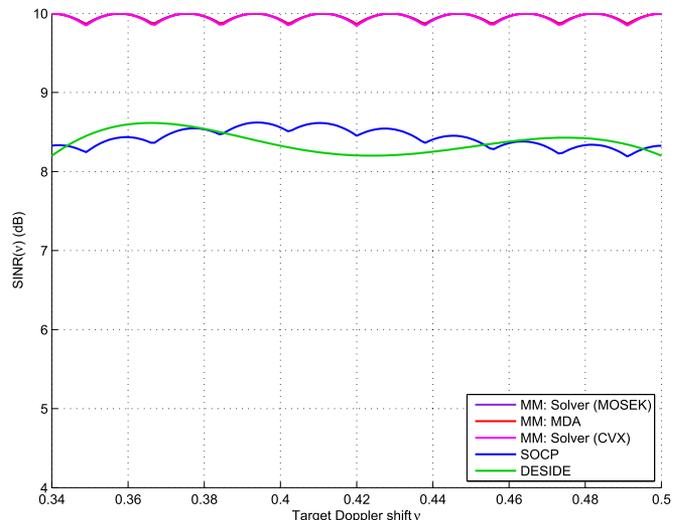
Fig. 6. Robust designs versus non-robust designs under different PAR levels and noise.

with

$$\mathbf{H}(\nu) = \text{Diag} \left(\left[1, e^{j2\pi\nu}, \dots, e^{j2\pi(N-1)\nu} \right]^T \right), \quad (69)$$

and all the other parameters follow Section V-A1. The variable \mathbf{s} is derived from optimization; once \mathbf{s} is known, the optimal $\{\mathbf{w}_i\}$ is also known (cf. (17)). The reason for using $\text{SINR}(\nu)$ is related to the detection mechanism of the filter bank: once the received signal is passed through the filter bank, we pick the largest SINR to compare with a predetermined threshold for detection; when the target Doppler is actually ν (still falling within the uncertainty interval), the largest SINR for threshold comparison is thus expressed as $\text{SINR}(\nu)$ and in the performance evaluation, we want $\text{SINR}(\nu)$ to be as large as possible. In Figure 6, we carry out a comparison between robust and non-robust designs under different PAR levels and noise. Under both white noise and colored noise, the robust design has a much smaller scale of fluctuation than its non-robust counterpart. Although the non-robust design achieves slightly higher SINR in a small neighborhood around the nominal value, its worst-case performance across the interval can be arbitrarily bad. Moreover, when imposing different levels of the PAR constraint, we see no significant change of $\text{SINR}(\nu)$ in the robust design, while the opposite is the case with the non-robust design.

3) *Comparison with Existing Methods:* Several existing robust designs have been proposed in the open literature. We compare the proposed two algorithms with the existing DESIDE [10] and the SOCP-based algorithm in [13]. In order to gain more insight, we include the in-built Matlab nonconvex optimization solver, namely, Fmincon, as an additional benchmark if it is applicable. To enable fair comparison, only the energy constraint


 Fig. 7. $\text{SINR}(\nu)$ versus Doppler shift ν for four methods: MM: MDA, MM: Solver (proposed methods) and SOCP, DESIDE (benchmark methods).

is enforced, i.e., $\rho = N$, and all four methods initiate from the same code $\mathbf{s}^{(0)}$.

First, we set $\mathbf{s}^{(0)}$ to be the generalized Barker code (with unit energy). Since the Fmincon solver fails to provide a feasible solution when initializing from this known sequence, Fmincon is not applicable here and its performance is not displayed. In Figure 7, we plot $\text{SINR}(\nu)$ in the uncertainty interval for the four methods. Our proposed methods achieve a worst-case SINR (the smallest value across the uncertainty interval, i.e., $\min_{\nu \in \Omega_T} \text{SINR}(\nu)$) of around 9.8 dB, while both benchmark

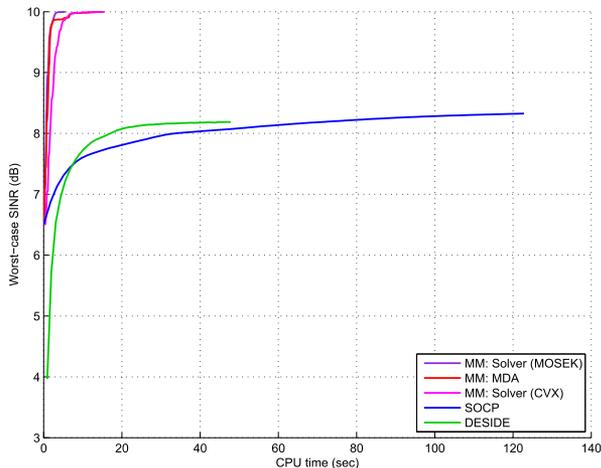


Fig. 8. Objective function value versus CPU time for four methods: MM: MDA, MM: Solver (proposed methods) and SOCP, DESIDE (benchmark methods).

algorithms achieve around 8.2 dB. In Figure 8, we plot the objective function value versus CPU time for the four methods. The four methods have different objective functions, so we focus on the convergence speed. Our proposed algorithms converge within 15 seconds, while SOCP needs more than 120 seconds and DESIDE needs about 48 seconds. So in this particular instance, the proposed algorithms are superior in terms of both worst-case SINR and convergence speed.

Now we set $\mathbf{s}^{(0)}$ to be a random code with constant modulus and unit energy, and generate 100 starting points. All the reported performances are averaged over the 100 instances. When the initialization is random, the performance of Fmincon gets better, and we should discuss its performance. The worst-case SINR is evaluated as $\min_{\nu \in \Omega_T} \text{SINR}(\nu)$. We vary the sequence length N among $\{20, 30, 40\}$ and the number of filters I among $\{N - 15, N - 10\}$ for a particular N . We present the results in Table I. In terms of worst-case SINR, Solver-based MM (either CVX or MOSEK) achieves the best performance; it beats MDA-based MM, DESIDE, SOCP, and Fmincon by around 0.10 dB, 1.76 dB, 0.82 dB, and 2.44 dB, respectively. Solver-based MM slightly beats MDA-based MM due to the relatively inexact solution to the subproblem caused by MDA-based MM. The achieved worst-case SINR of Fmincon is the lowest among all the methods since it is merely a general nonconvex optimization solver and cannot solve this specific problem quite well. DESIDE achieves the second lowest worst-case SINR because of the single receiving filter design. In terms of CPU time, with Fmincon excluded, the proposed MM algorithms beat the rest of the benchmarks: the fastest two, MDA-based MM and Solver-based MM (MOSEK), are about one order of magnitude faster than the slowest one, SOCP. Moreover, the performance of MDA-based MM and Solver-based MM (MOSEK) become more impressive for large N . When $N = 40$, MDA-based MM and Solver-based MM (MOSEK) are twice, three times, and ten times as fast as Solver-based MM (CVX), DESIDE, and SOCP, respectively. The underlying reason for the fast convergence speed is that MDA-based MM and Solver-based MM (MOSEK) do not use the CVX toolbox. From the perspective of problem size, we also see that for a fixed N , incorporat-

ing more filters (i.e., increasing I) can improve the worst-case SINR. This makes sense because we provide guarantee on more Doppler shift values in the uncertainty interval.

B. Multiuser Communications: Synchronous DS-CDMA

Experiment Settings: We consider a reverse-link synchronous DS-CDMA system with $I = 10$ users. The transmitting sequence length is $N = 20$ by default. We set $\alpha_i = \alpha = 1, \forall i$. The background noise covariance matrix \mathbf{R} is $\sigma^2 \mathbf{I}$ (white noise), where $\sigma^2 = 5 \times 10^{-3}$. The channel matrix \mathbf{H}_i is

$$\mathbf{H}_i = h_{i,1} \mathbf{I} + h_{i,2} \mathbf{J}_1, \forall i, \quad (70)$$

which means we consider two paths with different gains, with $h_{i,1} \sim \mathcal{CN}(0, 0.9)$ and $h_{i,2} \sim \mathcal{CN}(0, 0.1)$ being independent complex Gaussian random variables, and \mathbf{J}_1 is a Toeplitz matrix with the first subdiagonal entries being 1 and 0 elsewhere. As for the constraint set, we set $\rho = 1, 1 + N/200, N$ for performance comparison. Here the off-the-shelf is only specified as CVX because we find the Fusion Matlab API in MOSEK fails to solve the convex maximization problem well in this particular application. We observe violation of feasibility and suboptimality compared with CVX.

1) *Monotonic Property of the Proposed Algorithms:* We implement our proposed algorithms, Solver-based MM and MDA-based MM. Both algorithms initiate from the same random sequences $\{\mathbf{s}_i^{(0)}\}_{i=1}^I$. In Figure 9, we present the monotonic property of the proposed algorithms. The worst-case SINR (i.e., the objective function value) is monotonically increasing as time passes. When we decrease the parameter ρ from N to 1, the optimized worst-case SINR decreases because the constraint set becomes more and more tightened. Here the two algorithms reach almost the same optimized value, but MDA-based MM converges faster than Solver-based MM. In terms of CPU time, MDA-based MM is 0.5 – 1.5 orders of magnitude faster.

2) *Performance Evaluation of the Proposed Algorithms:* Now we generate 20 Gaussian channel realizations and each start with 10 random starting sequences $\{\mathbf{s}_i^{(0)}\}_{i=1}^I$. For SINR measurement, we run the algorithms with 10 initializations for every particular channel realization and get the best result. Then we compute the average of the best performances over 20 channel realizations. For CPU time, the performance is averaged over the $20 \times 10 = 200$ instances. The worst-case SINR is evaluated as $\min_{i=1,2,\dots,I} \text{SINR}_i(\{\mathbf{s}_i\})$. We vary the sequence length N among $\{20, 50, 80\}$ and the number of users I among $\{5, 10\}$. We set $\rho = 1$, that is, we design constant modulus sequences $\{\mathbf{s}_i\}$. The early works [46]–[48] assumed the channel matrix to be \mathbf{I} , and another work [49] assumed the channel matrix to be diagonal. In both cases, there is no actual channel that convolves with the sequences. In [20], the authors studied a general form of channel matrix and various optimization metrics, but did not consider the PAR constraint. The Matlab in-built solver Fmincon provides infeasible solutions almost surely, so Fmincon is inapplicable here. Hence, we do not have benchmarks for our maximin design problem. We present the results in Table II. In terms of worst-case SINR, Solver-based MM is always slightly better than MDA-based MM, which may result from the inexact solution of the subproblem caused by MDA-based MM. In terms of CPU time, MDA-based MM is much better; it is 0.5 – 1 orders of magnitude faster than Solver-based MM.

TABLE I
PERFORMANCE EVALUATION OF THE PROPOSED ALGORITHMS (THE FIRST THREE FROM THE LEFT) AND BENCHMARKS (THE LATTER THREE FROM THE LEFT) WITH DIFFERENT VALUES OF N AND I

	Solver – based MM (CVX)	Solver – based MM (MOSEK)	MDA-based MM	DESIDE	SOCP	Fmincon
$N = 20, I = 5$	9.047	9.156	9.011	7.974	7.550	7.355
$N = 20, I = 10$	9.804	9.829	9.630	8.018	8.660	6.026
$N = 30, I = 15$	9.845	9.835	9.795	7.985	9.390	7.129
$N = 30, I = 20$	9.902	9.880	9.855	7.997	9.261	7.471
$N = 40, I = 25$	9.882	9.880	9.775	7.995	9.288	8.070
$N = 40, I = 30$	9.935	9.859	9.772	7.909	9.351	7.760

(a) Performance evaluation: worst-case SINR (dB)

	Solver – based MM (CVX)	Solver – based MM (MOSEK)	MDA-based MM	DESIDE	SOCP	Fmincon
$N = 20, I = 5$	11.237	5.444	13.907	41.142	67.702	1.866
$N = 20, I = 10$	16.733	13.529	10.893	46.332	156.876	3.097
$N = 30, I = 15$	32.433	38.760	28.805	76.027	309.935	5.600
$N = 30, I = 20$	31.935	16.096	38.121	84.875	405.862	7.174
$N = 40, I = 25$	53.123	32.929	34.246	107.670	304.279	11.713
$N = 40, I = 30$	64.526	28.870	29.621	97.426	352.315	14.042

(b) Performance evaluation: CPU time (sec)

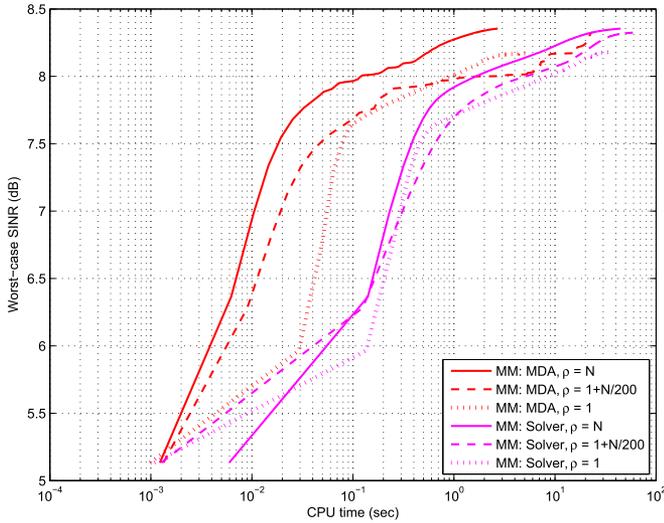


Fig. 9. Convergence plot: worst-case SINR versus CPU time; $N = 20$.

TABLE II
PERFORMANCE EVALUATION OF THE MM ALGORITHMS WITH DIFFERENT VALUES OF N AND I

	Solver-based MM		MDA-based MM	
	Worst – case SINR (dB)	CPU time (sec)	Worst – case SINR (dB)	CPU time (sec)
$N = 20, I = 5$	13.794	33.319	13.646	3.663
$N = 20, I = 10$	11.558	53.293	11.542	8.314
$N = 50, I = 5$	16.541	44.167	16.543	12.654
$N = 50, I = 10$	13.566	96.591	13.564	23.600
$N = 80, I = 5$	15.393	57.170	15.392	17.843
$N = 80, I = 10$	13.881	121.633	13.881	40.657

From the perspective of problem size, we can see an increase in CPU time with the growth of N and I . Also, we observe that the worst-case SINR of $I = 5$ is always a few dBs (seemingly 1.5 dB–3 dB from the table) higher than that of $I = 10$. This is

because the inclusion of more users brings more interference and thus reduces the chance of achieving a higher worst-case SINR.

VII. CONCLUSION

In this paper, we have proposed two algorithms based on the MM method to efficiently conduct the joint design of transmitting sequence(s) and receiving filters via maximin optimization. We have given an introduction of the vanilla MM method and elaborated its maximin extension, where the objective takes a pointwise minimum format. The algorithmic framework of the MM method for solving the maximin problem has been provided, and we have looked into some specific applications and examples as case studies. Numerical simulations have been presented based on these cases. The simulation results have shown that the proposed MM algorithms, both Solver-based and MDA-based, achieve higher objective values as well as a faster convergence speed compared with the benchmarks.

APPENDIX A

PROOF OF LEMMA 3

Proof: Referring to the expression of $\Sigma_I(\mathbf{s})$ from (4), one term in (23) can be further specified:

$$\begin{aligned}
 & -\text{Tr} \left(\mathbf{a}_i \mathbf{a}_i^H \cdot \left(\Sigma_I(\mathbf{s}) - \Sigma_I(\mathbf{s}^{(n)}) \right) \right) \\
 & = -\text{Tr} \left(\mathbf{a}_i \mathbf{a}_i^H \cdot \sum_j \beta_j \mathbf{M}_j \left(\mathbf{s} \mathbf{s}^H - \mathbf{s}^{(n)} \mathbf{s}^{(n)H} \right) \mathbf{M}_j^H \right) \\
 & = -\mathbf{s}^H \mathbf{A}_i \mathbf{s} + \left(\mathbf{s}^{(n)} \right)^H \mathbf{A}_i \mathbf{s}^{(n)}, \tag{71}
 \end{aligned}$$

where \mathbf{A}_i follows (28). Thus,

$$\begin{aligned}
 \text{SINR}_i(\mathbf{s}) & \geq \text{SINR}_i(\mathbf{s}^{(n)}) + 2\text{Re} \left[\mathbf{b}_i^H \left(\mathbf{s} - \mathbf{s}^{(n)} \right) \right] \\
 & \quad - \mathbf{s}^H \mathbf{A}_i \mathbf{s} + \left(\mathbf{s}^{(n)} \right)^H \mathbf{A}_i \mathbf{s}^{(n)}. \tag{72}
 \end{aligned}$$

We further minorize $\text{SINR}_i(\mathbf{s})$ by applying $\mathbf{s}^H \mathbf{A}_i \mathbf{s} \leq (\mathbf{s}^{(n)})^H \mathbf{A}_i \mathbf{s}^{(n)} + 2\text{Re}[\mathbf{s}^{(n)H} \mathbf{A}_i (\mathbf{s} - \mathbf{s}^{(n)})] + \lambda_{\max}(\mathbf{A}_i) \|\mathbf{s} - \mathbf{s}^{(n)}\|_2^2$:

$$\text{SINR}_i(\mathbf{s}) \geq \text{SINR}_i(\mathbf{s}^{(n)}) + 2\text{Re}[\mathbf{c}_i^H (\mathbf{s} - \mathbf{s}^{(n)})] - \lambda_{u,i} \|\mathbf{s} - \mathbf{s}^{(n)}\|_2^2 = \overline{\text{SINR}}_i(\mathbf{s}, \mathbf{s}^{(n)}), \quad (73)$$

where \mathbf{c}_i and $\lambda_{u,i}$ are defined in (27) and (29), respectively. ■

APPENDIX B PROOF OF LEMMA 4

Proof: If we relax \mathcal{S} to be $\mathcal{S}_{\text{relaxed}}$, then the problem becomes (same as (38))

$$\max_{\mathbf{s} \in \mathcal{S}_{\text{relaxed}}} \min_{\mathbf{p} \in \mathcal{P}} 2\text{Re} \left[\left((\mathbf{C} + \mathbf{s}^{(l)} \boldsymbol{\lambda}_u^T) \mathbf{p} \right)^H \mathbf{s} \right] + \mathbf{p}^T \mathbf{d}. \quad (74)$$

We observe that the objective function is concave-convex in \mathbf{s} and \mathbf{p} , and $\mathcal{S}_{\text{relaxed}}$ and \mathcal{P} are both nonempty compact convex sets. Following the results of [43, Corollary 37.6.2] and [43, Lemma 36.2], a saddle point exists for the relaxed problem.

Now we claim the saddle point of the relaxed problem, denoted by $(\mathbf{s}^*, \mathbf{p}^*)$, must satisfy $\mathbf{s}^* \in \mathcal{S}$, or equivalently $\|\mathbf{s}^*\|_2 = 1$. The argument is given by contradiction. Suppose $\|\mathbf{s}^*\|_2 < 1$. We can always find some element of \mathbf{s}^* , say \mathbf{s}_j^* , such that $|\mathbf{s}_j^*| < \sqrt{\frac{P}{N}}$. If not, then $\|\mathbf{s}^*\|_2 \geq \sqrt{(\sqrt{\frac{P}{N}})^2 \times N} = \sqrt{P} \geq 1$, causing contradiction. Then we reset the phase of \mathbf{s}_j^* to be aligned with the j th element of $(\mathbf{C} + \mathbf{s}^{(l)} \boldsymbol{\lambda}_u^T) \mathbf{p}^*$ and increase its modulus by a small amount without violating feasibility. The objective can be pushed up from the side of \mathbf{s} , causing contradiction with the saddle point nature of \mathbf{s}^* . The j th element of $(\mathbf{C} + \mathbf{s}^{(l)} \boldsymbol{\lambda}_u^T) \mathbf{p}^*$ has been assumed to be nonzero for simplicity. In case it becomes zero, the optimal solution of \mathbf{s}_j may be non-unique (and thus the saddle point is non-unique), but we can always find one on the boundary by properly increasing the modulus of the currently obtained \mathbf{s}_j^* if necessary.

Since the saddle point (or at least one saddle point) of the relaxed problem naturally satisfies $\mathbf{s}^* \in \mathcal{S}$ and $\mathbf{p}^* \in \mathcal{P}$, there must exist a saddle point for problem (35), and the saddle point can be obtained from solving the relaxed problem. ■

REFERENCES

- [1] M. Soltanalian, B. Tang, J. Li, and P. Stoica, "Joint design of the receive filter and transmit sequence for active sensing," *IEEE Signal Process. Lett.*, vol. 20, no. 5, pp. 423–426, May 2013.
- [2] J. Tropp, I. S. Dhillon, R. W. Heath Jr., and T. Strohmer, "Designing structured tight frames via an alternating projection method," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 188–209, Jan. 2005.
- [3] A. De Maio, Y. Huang, M. Piezzo, S. Zhang, and A. Farina, "Design of optimized radar codes with a peak to average power ratio constraint," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2683–2697, Jun. 2011.
- [4] P. Stoica, H. He, and J. Li, "Optimization of the receive filter and transmit sequence for active sensing," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1730–1740, Apr. 2012.
- [5] P. Stoica, J. Li, and Y. Xie, "On probing signal design for MIMO radar," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4151–4161, Aug. 2007.
- [6] C.-Y. Chen and P. Vaidyanathan, "MIMO radar waveform optimization with prior information of the extended target and clutter," *IEEE Trans. Signal Process.*, vol. 57, no. 9, pp. 3533–3544, Sep. 2009.
- [7] B. Friedlander, "Waveform design for MIMO radars," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 3, pp. 1227–1238, Jul. 2007.
- [8] A. De Maio, Y. Huang, and M. Piezzo, "A Doppler robust max-min approach to radar code design," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4943–4947, Sep. 2010.
- [9] A. Aubry, A. De Maio, M. Piezzo, and A. Farina, "Radar waveform design in a spectrally crowded environment via nonconvex quadratic optimization," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 2, pp. 1138–1152, Apr. 2014.
- [10] M. M. Naghsh, M. Soltanalian, P. Stoica, M. Modarres-Hashemi, A. De Maio, and A. Aubry, "A Doppler robust design of transmit sequence and receive filter in the presence of signal-dependent interference," *IEEE Trans. Signal Process.*, vol. 62, no. 4, pp. 772–785, Feb. 2014.
- [11] M. F. Hanif, L.-N. Tran, A. Tolli, and M. Juntti, "Computationally efficient robust beamforming for SINR balancing in multicell downlink with applications to large antenna array systems," *IEEE Trans. Commun.*, vol. 62, no. 6, pp. 1908–1920, Jun. 2014.
- [12] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Rev.*, vol. 38, no. 1, pp. 49–95, 1996.
- [13] A. Aubry, A. De Maio, and M. M. Naghsh, "Optimizing radar waveform and Doppler filter bank via generalized fractional programming," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 8, pp. 1387–1399, Dec. 2015.
- [14] M. A. Richards, J. A. Scheer, W. A. Holm, and W. L. Melvin, *Principles of Modern Radar*. Citeseer, 2010.
- [15] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret, "Applications of second-order cone programming," *Linear Algebra Appl.*, vol. 284, no. 1, pp. 193–228, 1998.
- [16] G. Cui, H. Li, and M. Rangaswamy, "MIMO radar waveform design with constant modulus and similarity constraints," *IEEE Trans. Signal Process.*, vol. 62, no. 2, pp. 343–353, Jan. 2014.
- [17] J. Li and P. Stoica, "MIMO radar with colocated antennas," *IEEE Signal Process. Mag.*, vol. 24, no. 5, pp. 106–114, Sep. 2007.
- [18] H. Xu, R. S. Blum, J. Wang, and J. Yuan, "Colocated MIMO radar waveform design for transmit beampattern formation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 2, pp. 1558–1568, Apr. 2015.
- [19] S. M. Karbasi, A. Aubry, V. Carotenuto, M. M. Naghsh, and M. H. Bastani, "Knowledge-based design of space-time transmit code and receive filter for a multiple-input-multiple-output radar in signal-dependent interference," *Radar, Sonar Navig.*, vol. 9, no. 8, pp. 1124–1135, 2015.
- [20] D. P. Palomar, J. M. Cioffi, and M. A. Lagunas, "Joint Tx-Rx beamforming design for multicarrier MIMO channels: A unified framework for convex optimization," *IEEE Trans. Signal Process.*, vol. 51, no. 9, pp. 2381–2401, Sep. 2003.
- [21] W. Santipach, "Signature quantization in fading CDMA with limited feedback," *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 569–577, Feb. 2011.
- [22] V. K. Lau, "On the analysis of peak-to-average ratio (PAR) for IS95 and CDMA2000 systems," *IEEE Trans. Veh. Technol.*, vol. 49, no. 6, pp. 2174–2188, Nov. 2000.
- [23] J. Tropp, I. S. Dhillon, R. W. Heath Jr., and T. Strohmer, "CDMA signature sequences with low peak-to-average-power ratio via alternating projection," in *Proc. IEEE Conf. Rec. 37th Asilomar Conf. Signals, Syst. Comput.*, 2003, vol. 1, pp. 475–479.
- [24] D. W. Cai, T. Q. Quek, C. W. Tan, and S. H. Low, "Max-min SINR coordinated multipoint downlink transmission: duality and algorithms," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5384–5395, Oct. 2012.
- [25] D. W. Cai, T. Q. Quek, and C. W. Tan, "A unified analysis of max-min weighted SINR for MIMO downlink system," *IEEE Trans. Signal Process.*, vol. 59, no. 8, pp. 3850–3862, Aug. 2011.
- [26] E. Karipidis, N. D. Sidiropoulos, and Z.-Q. Luo, "Quality of service and max-min fair transmit beamforming to multiple cochannel multicast groups," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1268–1279, Mar. 2008.
- [27] M. Soltanalian, A. Gharanjik, and M. B. Shankar, "Grab-n-pull: An optimization framework for fairness-achieving networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 3301–3305.
- [28] S. X. Wu, A. M.-C. So, J. Pan, and W.-K. Ma, "Semidefinite relaxation and approximation analysis of a beamformed alamouti scheme for relay beamforming networks," arXiv:1603.05680, 2016.
- [29] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [30] G. Scutari, F. Facchinei, L. Lampariello, P. Song, and S. Sardellitti, "Parallel and distributed methods for nonconvex optimization—Part II: Applications," arXiv:1601.04059, 2016.
- [31] J. Song, P. Babu, and D. P. Palomar, "Sequence set design with good correlation properties via majorization-minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 11, pp. 2866–2879, Jun. 2016.
- [32] Z. Wang, P. Babu, and D. P. Palomar, "Design of PAR-constrained sequences for MIMO channel estimation via majorization-minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6132–6144, Aug. 2011.
- [33] J.-S. Pang, M. Razaviyayn, and A. Alvarado, "Computing B-stationary points of nonsmooth DC programs," arXiv:1511.01796, 2015.
- [34] J. Pang, "Partially B-regular optimization and equilibrium problems," *Math. Oper. Res.*, vol. 32, no. 3, pp. 687–699, 2007.

- [35] A. Beck and M. Teboulle, "Mirror descent and nonlinear projected subgradient methods for convex optimization," *Oper. Res. Lett.*, vol. 31, no. 3, pp. 167–175, 2003.
- [36] A. Hjørungnes, *Complex-Valued Matrix Derivatives: With Applications in Signal Processing and Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2011.
- [37] A. Aubry, A. De Maio, M. Piezzo, M. M. Naghsh, M. Soltanalian, and P. Stoica, "Cognitive radar waveform design for spectral coexistence in signal-dependent interference," in *Proc. IEEE Radar Conf.*, 2014, pp. 0474–0478.
- [38] A. Aubry, A. DeMaio, A. Farina, and M. Wicks, "Knowledge-aided (potentially cognitive) transmit signal and receive filter design in signal-dependent clutter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 1, pp. 93–117, Jan. 2013.
- [39] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *Amer. Statist.*, vol. 58, no. 1, pp. 30–37, 2004.
- [40] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 1126–1153, 2013.
- [41] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, vol. 317. New York, NY, USA: Springer, 2009.
- [42] T. Lipp and S. Boyd, "Variations and extension of the convex–concave procedure," *Optim. Eng.*, vol. 17, no. 2, pp. 263–287, Jun. 2016.
- [43] R. T. Rockafellar, *Convex Analysis*. Princeton, NJ, USA: Princeton Univ. Press, 1970.
- [44] *The Mosek Optimization Toolbox for MATLAB Manual, Version 7.1* (revision 28). [Online]. Available: <http://mosek.com>, accessed on: Mar. 20, 2015.
- [45] M. Grant and S. Boyd, *CVX: MATLAB Software for Disciplined Convex Programming, Version 2.1*. [Online]. Available: <http://cvxr.com/cvx>, Mar. 2014.
- [46] P. Viswanath and V. Anantharam, "Optimal sequences for CDMA under colored noise: A Schur–Saddle function property," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1295–1318, Jun. 2002.
- [47] S. Ulukus and R. D. Yates, "Iterative construction of optimum signature sequence sets in synchronous CDMA systems," *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 1989–1998, Jul. 2001.
- [48] S. Ulukus and A. Yener, "Iterative transmitter and receiver optimization for CDMA networks," *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 1879–1884, Nov. 2004.
- [49] S. Ulukus and R. D. Yates, "Optimum multiuser detection is tractable for synchronous CDMA systems using m-sequences," *IEEE Commun. Lett.*, vol. 2, no. 4, pp. 89–91, Apr. 1998.
- [50] S. Ying, P. Babu, and D. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Trans. Signal Process.*, 2016

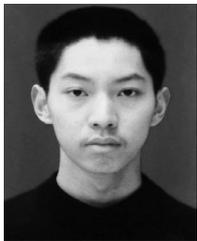


Daniel P. Palomar (S'99–M'03–SM'08–F'12) received the electrical engineering and Ph.D. degrees from the Technical University of Catalonia, Barcelona, Spain, in 1998 and 2003, respectively.

He is a Professor in the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology (HKUST), Hong Kong, which he joined in 2006. He had previously held several research appointments, namely, at King's College London, London, U.K.; Stanford University, Stanford, CA, USA; Telecommunications Technological Center of Catalonia, Barcelona; Royal Institute of Technology, Stockholm, Sweden; University of Rome "La Sapienza," Rome, Italy; and Princeton University, Princeton, NJ, USA. His current research interests include applications of convex optimization theory, game theory, and variational inequality theory to financial systems, big data systems, and communication systems.

Dr. Palomar is a Fellow of the Institute for Advance Study, HKUST, since 2013. He is an IEEE Fellow and received a 2004/06 Fulbright Research Fellowship, the 2004 and 2015 (co-author) Young Author Best Paper Awards by the IEEE Signal Processing Society, the 2015–16 HKUST Excellence Research Award, the 2002/03 best Ph.D. prize in Information Technologies and Communications by the Technical University of Catalonia, the 2002/03 Rosina Ribalta first prize for the Best Doctoral Thesis in Information Technologies and Communications by the Epson Foundation, and the 2004 prize for the best Doctoral Thesis in Advanced Mobile Communications by the Vodafone Foundation and COIT.

He is a Guest Editor of the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING 2016 Special Issue on "Financial Signal Processing and Machine Learning for Electronic Trading" and has been an Associate Editor of the IEEE TRANSACTIONS ON INFORMATION THEORY and the IEEE TRANSACTIONS ON SIGNAL PROCESSING, a Guest Editor of the *IEEE Signal Processing Magazine* 2010 Special Issue on "Convex Optimization for Signal Processing," the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS 2008 Special Issue on "Game Theory in Communication Systems," and the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS 2007 Special Issue on "Optimization of MIMO Transceivers for Realistic Communication Networks."



Licheng Zhao received the B.S. degree in information engineering from Southeast University, Nanjing, China, in 2014. He is currently working toward the Ph.D. degree with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology (HKUST), Hong Kong. His research interests include optimization theory and fast algorithms, with applications in signal processing, machine learning, and financial engineering.