Sparse Blind Deconvolution and Demixing via Block Majorization-Minimization

Mengting Chen and Ziping Zhao
ShanghaiTech University, Shanghai
{chenmt, zipingzhao}@shanghaitech.edu.cn

***Abstract*—To support vastly growing intelligent devices, massive connectivity with low-latency communications has become a critical requirement. In this paper, we consider a multiple-input multiple-output network, where a large amount of devices are connected to an access point sporadically. We aim to simultaneously detect the active devices and recover the transmitted signals from the received mixed measurements, without a priori channel information. The problem is mathematically modeled based on the idea of blind deconvolution and demixing for sparse signals. We formulate the optimization problem via nonconvex matrix factorization, and propose an efficient block majorization-minimization algorithm, where the signals and filters are updated with analytical solutions in an alternating way. The proposed algorithm has much lower per-iteration computational complexity compared to state-of-the-art algorithms and hence is more scalable to large-size problems. Numerical results demonstrate that our method is able to recover the sparse signals and filters with higher precision as well as faster convergence in comparison with existing methods.**

I. Introduction

With the explosion of small and affordable computing devices, the capability of connecting a vast number of devices, or massive device connectivity, has become an essential requirement for future wireless communications. However, since the resources in a cellular network are limited, the devices are designed to keep inactive most of the time unless triggered by external event, leading to a bursty and sporadic signal transmission pattern. For example, in the mobile networks, applications running in the background of smartphones or tablets can periodically send or receive data. In the IoT networks, energy-constrained devices often remain idle to conserve power and only become active intermittently to send data. Furthermore, emerging applications in 5G, such as virtual reality, real-time video conferencing, online gaming, etc., require information to be delivered with ultra-low latency [1], [2]. It is thus urgent to support a communications network with both massive connectivity and low latency.

Massive connectivity in communication mainly has three key challenges. In scenarios with numerous devices, only a subset may need to transmit data intermittently. Therefore, it is necessary to identify active devices at a given time to help manage network resources to minimize idle communication overhead. Once active devices are identified, the system must estimate the channel conditions for each device, which is crucial for coding strategies and thus reduces signal interference. Finally, transmitted data from the active devices need to be accurately recovered. While the three challenges can be addressed separately, a more effective approach is to jointly detect active devices and estimate channel information [3], [4], followed by data recovery [5]. However, this two-stage procedure still introduces additional latency, since it requires channel information estimation for data recovery. To reduce channel signaling overhead, it could be more desirable to simultaneously detect active devices and recovery the transmitted data without knowing channel information [6]. This target can be mathematically modeled based on the idea of blind deconvolution and demixing for sparse signals (SBD^2).

Specifically, the SBD^2 for joint active devices detection and transmitting data recovery relies on a bilinear model with a group-sparse assumption on the transmitting signal, which is introduced for active device detection. Since the bilinear structure essentially forms a low-rank matrix [7], [8], the SBD^2 problem can be formulated based on a sparse penalty [9] and a low-rank penalty [10]. A convex approach was proposed as a semidefinite programming (SDP) problem. However, SDP is computationally prohibitive especially for large-scale problems [11]. In [12], the authors used difference-of-convex-functions [13] to represent the rank-one property. An iteratively reweighted SDP (IR-SDP) method was then proposed for problem solving. In each iteration, an SDP is solved. Since solving an SDP needs to scale the problem to a higher-dimensional variable space, both SDP and IR-SDP are computational expensive, especially for large-scale problems.

In this paper, we consider an SBD^2 problem arising in a multiple-input multiple-output network. The problem is formulated based on nonconvex matrix factorization. We propose a block majorization-minimization (BMM) algorithm [14]–[16] to address the nonconvex problem. In each iteration, the signals and the filters are updated with closed-form solutions. Consequently, BMM exhibits significantly lower per-iteration computational complexity compared to SDP and IR-SDP, making it more suitable for problems of large sizes. We also prove that the proposed BMM algorithm is equivalent to an alternating proximal gradient descent algorithm. Numerical simulations verify that BMM recovers the unknown sparse signals and the filters with higher precision and faster convergence than the existing methods.

II. Problem Formulation

We consider a multiple-input multiple-output (MIMO) communication system, where S devices with a single antenna are connected to a base station with R antennas; see Fig. 1. We

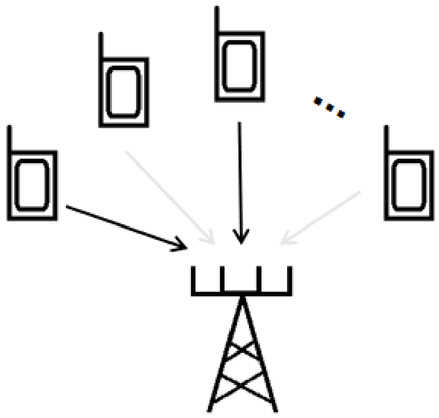

Fig. 1: In a MIMO system, multiple devices connect to a BS with several antennas. Black arrows show active uplink transmissions, while grey lines indicate inactive connections.

consider a block-fading model where channels are independent quasi-static flat-fading in each coherence time. In the massive access scenario, only a small fraction of S devices are active and access the base station in a coherence block. In the uplink transmission, let $\mathbf{x}_k \in \mathbb{C}^N$ be the data sequence of length N from the device k ($k = 1, \ldots, S$). When device k is active, $\mathbf{x}_k \neq \mathbf{0}$; otherwise, $\mathbf{x}_k = \mathbf{0}$. The signal sequence transmitted by the device k over M time slots is $\mathbf{A}_k\mathbf{x}_k$, where $\mathbf{A}_k \in \mathbb{C}^{M\times N}$ is a preassigned encoding matrix available to the base station. The signal $\mathbf{A}_k\mathbf{x}_k$ is passing through a channel $\mathbf{h}_{rk} \in \mathbb{C}^L$ to antenna r ($r = 1, \ldots, R$), where $L < M$ is the maximal delay spread of the finite impulse response. Denote $\mathring{\mathbf{h}}_{rk} = [\mathbf{h}_{rk}^\mathsf{T}, \mathbf{0}]^\mathsf{T} \in \mathbb{C}^M$ as the zero-padded filter. The transmission process can be represented by a circular convolution between $\mathbf{A}_k\mathbf{x}_k$ and $\mathring{\mathbf{h}}_{rk}$. Then, the signal received by the antenna r at the base station is

$$\mathbf{y}_{\circledast,r} = \sum_{k=1}^{S} (\mathbf{A}_k\mathbf{x}_k) \circledast \mathring{\mathbf{h}}_{rk} + \mathbf{e}_{\circledast,r}, \tag{1}$$

where $\circledast$ denotes the circular convolution and $\mathbf{e}_{\circledast,r}$ is an additive noise. Denote $\mathbf{F}$ as the M-point unitary discrete Fourier transform matrix. The received signal in frequency domain is

$$\mathbf{y}_r := \mathbf{F}\mathbf{y}_{\circledast,r} = \sum_{k=1}^{S} \sqrt{M}\,(\mathbf{F}\mathbf{A}_k\mathbf{x}_k) \odot (\mathbf{B}\mathbf{h}_{rk}) + \mathbf{e}_r, \tag{2}$$

where $\odot$ denotes the Hadamard product, $\mathbf{B}$ contains the first L columns of $\mathbf{F}$, and $\mathbf{e}_r = \mathbf{F}\mathbf{e}_{\circledast,r}$.

Remark: In practice, the encoding matrix $\mathbf{A}_k$ is often designed to have a convenient structure which can lead to fast computations and minimal memory requirements. For example, it can be chosen as $\mathbf{A}_k = \mathsf{Diag}(\mathbf{q}_k)\,\mathbf{P}$, where $\mathbf{q}_k \in \mathbb{C}^M$ is a vector with entries being ± 1 and $\mathbf{P} \in \mathbb{C}^{M\times N}$ is a partial matrix containing the N columns of an $M \times M$ Hadamard/Fourier matrix [17].

Our goal is to recover nonzero signals $\mathbf{x}_k$'s of active devices from the observations $\{\mathbf{y}_r\}_{r=1}^R$. To reduce the channel signaling overhead, we assume that $\mathbf{h}_{rk}$'s are not available to both receivers and transmitters. Then, the problem is to simultaneously recover the message signals $\mathbf{x}_k$'s and the channels $\mathbf{h}_{rk}$'s from $\mathbf{y}_r$'s. Mathematically, the task of interest can be modeled as an SBD^2 problem.

Define

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_1 & \cdots & \mathbf{y}_R \end{bmatrix}, \tag{3}$$

and

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{h}_{1k} & \cdots & \mathbf{h}_{Rk} \end{bmatrix}, \tag{4}$$

for $k = 1, \ldots, S$. We propose a nonconvex optimization problem for SBD^2 in ($\star$),[1] where the objective is the combination of a fitting-error term and a group sparsity penalty $\sum_{k=1}^S \|\mathbf{x}_k\|$ with γ a tuning parameter. It can be easily verified that in model (2), each signal and filter pair $\{\mathbf{x}_k, \mathbf{h}_{rk}\}$ in model can be only identified up to a scaling factor; i.e., $\{\mathbf{x}_k^\star, \mathbf{h}_{rk}^\star\}$ and $\{\alpha\mathbf{x}_k^\star, \alpha^{-1}\mathbf{h}_{rk}^\star\}$ for any $\alpha \neq 0$ lead to the same output. To alleviate such ambiguity, we restrict the filters $\mathbf{h}_{rk}$ to have unit length in ($\star$) without affecting the optimality.

III. The Block Majorization-Minimization Method

Block majorization-minimization (BMM) [14], [16] can be viewed as a judicious combination of the block coordinate descent (BCD) method [18] and the majorization minimization (MM) method [15]. The BCD method aims to find a local optimal solution by optimizing the objective along one variable block each time and solving for different blocks successively. One potential limitation of BCD is the requirement that each subproblem needs to be solved exactly, which could be difficult like for nonconvex subproblems. BMM removes the above restriction by optimizing in each variable block a subproblem where the original objective function in BCD is replaced by a surrogate function which can lead to cheap iterations.

Specifically, given an optimization problem as follows:

$$\underset{\mathbf{x}\in\mathcal{X}}{\mathsf{minimize}} \quad f(\mathbf{x}), \tag{5}$$

[1] Throughout the paper, $\|\cdot\|$ is used to denote the ℓ_2-norm for vectors or the Frobenius norm for matrices.

$$\begin{aligned} \underset{\{\mathbf{x}_k,\mathbf{H}_k\}_{k=1}^S}{\text{minimize}} \quad & \frac{1}{2}\left\| \sum_{k=1}^{S} \sqrt{M}\,((\mathbf{F}\mathbf{A}_k\mathbf{x}_k) \otimes \mathbf{1}) \odot (\mathbf{B}\mathbf{H}_k) - \mathbf{Y} \right\|^2 + \gamma \sum_{k=1}^{S} \|\mathbf{x}_k\| \\ \text{subject to} \quad & \|\mathbf{h}_{rk}\| = 1,\ k = 1,\ldots,S,\ r = 1,\ldots,R. \end{aligned} \tag{$\star$}$$

where $\mathcal{X} \subseteq \mathbb{R}^N$. Suppose the optimization variable $\mathbf{x}$ can be partitioned into I blocks as $\mathbf{x} \triangleq (\mathbf{x}_1, \ldots, \mathbf{x}_I)$ where $\mathbf{x}_i \in \mathcal{X}_i$ and $\mathcal{X} = \prod_{i=1}^I \mathcal{X}_i$ with $\mathcal{X}_i \subseteq \mathbb{R}^{n_i}$ and $\sum_{i=1}^I n_i = N$. At each iteration of BMM, one variable block, say, $\mathbf{x}_i$, is updated according to the following update rules:

$$\begin{cases} \mathbf{x}_i^+ \in \arg\min_{\mathbf{x}_i \in \mathcal{X}_i} \bar{f}_i(\mathbf{x}_i, \underline{\mathbf{x}}), \\ \mathbf{x}_{-i}^+ = \underline{\mathbf{x}}_{-i} \text{ with } \mathbf{x}_{-i} \triangleq (\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_I), \end{cases} \tag{6}$$

where $\bar{f}_i$ is an upperbound function for f_i with respect to variable $\mathbf{x}_i$, and $\mathbf{x}_i^+$ and $\mathbf{x}_{-i}^+$ are the newly updated variables. The algorithm is monotonic and iteratively runs until some convergence criterion is met. Choosing the surrogate functions $\bar{f}_i$'s is the crucial part in BMM algorithm development. Generally speaking, they could be derived in multiple ways. However, a properly chosen one by taking the specific problem structure into account will make the iterative updating steps cheap while maintaining a fast algorithm convergence over iterations. In practice, surrogate functions will be much applaudable if they will lead to analytical solutions for the blockwise subproblems.

IV. Proposed Algorithm

In this section, we propose a BMM-based algorithm to effectively solve the problem ($\star$). The matrix factorization structure inherent in ($\star$) allows the algorithm to iteratively update the signals and filters in an alternating fashion. Furthermore, by finding suitable surrogate functions that exploit the blockwise structure of the problem, the two variables are updated with analytical solutions.

A. Updating the Signals

With fixed $\underline{\mathbf{H}}_k$'s,[2] we derive the update rule for $\mathbf{x}_k$'s. Let $\mathbf{y} = \text{vec}(\mathbf{Y})$ and $\mathbf{x} = \left[\mathbf{x}_1^\mathsf{T}, \ldots, \mathbf{x}_S^\mathsf{T}\right]^\mathsf{T}$. Define a block matrix $\mathbf{C} \in \mathbb{C}^{MR \times NS}$ with $\mathbf{C}_{rk} = \sqrt{M}\text{Diag}\left(\mathbf{B}\underline{\mathbf{h}}_{rk}\right)\mathbf{F}\mathbf{A}_k \in \mathbb{C}^{M \times N}$ the block at the r-th row and the k-th column. Problem ($\star$) with respect to (w.r.t.) variable $\mathbf{x}$ can be written as

$$\min_{\mathbf{x}} \quad \frac{1}{2}\|\mathbf{C}\mathbf{x} - \mathbf{y}\|^2 + \gamma \sum_{k=1}^S \|\mathbf{x}_k\|. \tag{7}$$

To apply the BMM technique, we first introduce the following useful lemma.

Lemma 1. *[15] Let $\mathbf{L}$, $\mathbf{M}$ be $n \times n$ Hermitian matrices satisfying $\mathbf{M} - \mathbf{L} \succeq \mathbf{0}$. Then for any $\underline{\mathbf{x}} \in \mathbb{C}^n$, it follows that*

$$\mathbf{x}^\mathsf{H}\mathbf{L}\mathbf{x} \leq \mathbf{x}^\mathsf{H}\mathbf{M}\mathbf{x} - 2\text{Re}\left\{\mathbf{x}^\mathsf{H}\left(\mathbf{M} - \mathbf{L}\right)\underline{\mathbf{x}}\right\} + \underline{\mathbf{x}}^\mathsf{H}\left(\mathbf{M} - \mathbf{L}\right)\underline{\mathbf{x}},$$

where the equality is attained at $\mathbf{x} = \underline{\mathbf{x}}$.

The first term in the objective of (7) contains the quadratic term over $\mathbf{x}$, which is $\frac{1}{2}\mathbf{x}^\mathsf{H}\mathbf{C}^\mathsf{H}\mathbf{C}\mathbf{x}$. Based on Lemma 1, taking $\mathbf{C}^\mathsf{H}\mathbf{C}$ as $\mathbf{L}$, we can construct a surrogate (i.e., upper-bound) function for the quadratic term by choosing $\mathbf{M} = C\mathbf{I}$ where C is a scalar satisfying $C \geq \left\|\mathbf{C}^\mathsf{H}\mathbf{C}\right\|_2$. Here, $\left\|\mathbf{C}^\mathsf{H}\mathbf{C}\right\|_2$ is the spectral norm of $\mathbf{C}^\mathsf{H}\mathbf{C}$. A surrogate function for the fitting-error term can thus be designed as

$$\begin{aligned} &\frac{1}{2}\|\mathbf{C}\mathbf{x} - \mathbf{y}\|^2 \\ \leq &\sum_{k=1}^S \left(\frac{C}{2}\mathbf{x}_k^\mathsf{H}\mathbf{x}_k - \text{Re}\left\{\mathbf{m}_k^\mathsf{H}\mathbf{x}_k\right\}\right) + \text{const.} \\ = &\sum_{k=1}^S \frac{C}{2}\left\|\mathbf{x}_k - C^{-1}\mathbf{m}_k\right\|^2 + \text{const.}, \end{aligned} \tag{8}$$

where $\mathbf{m} = \left(C\mathbf{I} - \mathbf{C}^\mathsf{H}\mathbf{C}\right)\underline{\mathbf{x}} + \mathbf{C}^\mathsf{H}\mathbf{y}$ and $\mathbf{m}_k$ contains the $(N(k-1)+1, \ldots, Nk)$-th elements of $\mathbf{m}$ for $k = 1, \ldots, S$. Since the surrogate function in (8) has a decoupled structure over S devices, each $\mathbf{x}_k$ can be solved in a parallel way, leading to solve the following sub-problem

$$\underset{\mathbf{x}_k}{\text{minimize}} \quad \frac{1}{2}\left\|\mathbf{x}_k - C^{-1}\mathbf{m}_k\right\|^2 + C^{-1}\gamma\|\mathbf{x}_k\|, \tag{9}$$

which is a group soft-thresholding step [19] and the solution is given by

$$\mathbf{x}_k^+ = \left(1 - \gamma\left\|\mathbf{m}_k\right\|^{-1}\right)^+ C^{-1}\mathbf{m}_k, \tag{10}$$

where $(x)^+ := \max\{x, 0\}$.

B. Updating the Filters

In the filter-updating step, we first rewrite the problem in a more compact way. Denote $\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \vdots \\ \mathbf{H}_S \end{bmatrix} \in \mathbb{C}^{LS \times R}$. The subproblem w.r.t. $\mathbf{H}$ is a constrained least-squares problem:

$$\begin{aligned} &\underset{\mathbf{H}}{\text{minimize}} && \frac{1}{2}\|\mathbf{D}\mathbf{H} - \mathbf{Y}\|^2 \\ &\text{subject to} && \|\mathbf{h}_{rk}\| = 1, \ k = 1, \ldots, S, \ r = 1, \ldots, R, \end{aligned} \tag{11}$$

where $\mathbf{D} \in \mathbb{C}^{M \times LS}$ is a block matrix with its k-th block being $\mathbf{D}_k = \sqrt{M}\text{Diag}\left(\mathbf{F}\mathbf{A}_k\underline{\mathbf{x}}_k\right)\mathbf{B} \in \mathbb{C}^{M \times L}$.

Similar to last subsection, we aim to find a surrogate function for the objective in (11). Expanding the objective, we obtain a quadratic term $\frac{1}{2}\text{Tr}\left(\mathbf{H}^\mathsf{H}\mathbf{D}^\mathsf{H}\mathbf{D}\mathbf{H}\right)$. Based on Lemma 1, by choosing D such that $D\mathbf{I} \succeq \left\|\mathbf{D}^\mathsf{H}\mathbf{D}\right\|_2$, we can find a surrogate function of the quadratic term as follows:

$$\begin{aligned} &\frac{1}{2}\|\mathbf{D}\mathbf{H} - \mathbf{Y}\|^2 \\ \leq &\frac{D}{2}\mathbf{h}^\mathsf{H}\mathbf{h} - \text{Re}\left\{\mathbf{n}^\mathsf{H}\mathbf{h}\right\} + \text{cst.} \\ = &\sum_{r=1}^R \sum_{k=1}^S \frac{D}{2}\left\|\mathbf{h}_{rk} - D^{-1}\mathbf{n}_{rk}\right\|^2 + \text{cst.}, \end{aligned} \tag{12}$$

where $\mathbf{N} = \left(D\mathbf{I} - \mathbf{D}^\mathsf{H}\mathbf{D}\right)\underline{\mathbf{H}} + \mathbf{D}^\mathsf{H}\mathbf{Y} \in \mathbb{C}^{LS \times R}$, $\mathbf{h} = \text{vec}\left(\mathbf{H}\right) \in \mathbb{C}^{LSR}$, and $\mathbf{n}_{rk} \in \mathbb{C}^L$ is a subvector consisting of the $\left(L\left(rk-1\right)+1, \ldots, Lrk\right)$-th elements of $\mathbf{n} = \text{vec}\left(\mathbf{N}\right)$. Then, $\mathbf{H}$ can be found by solving SR subproblems,

$$\underset{\|\mathbf{h}_{rk}\|=1}{\text{minimize}} \quad \left\|\mathbf{h}_{rk} - D^{-1}\mathbf{n}_{rk}\right\|^2, \tag{13}$$

[2]In this paper, underlined variables denote those whose values are given.

By solving the Karush–Kuhn–Tucker conditions of (13), we derive its closed-form solution

$$\mathbf{h}_{rk}^{+} = \frac{\mathbf{n}_{rk}}{\|\mathbf{n}_{rk}\|}. \tag{14}$$

The overall algorithm is summarized in Algorithm 1.

Algorithm 1 BMM algorithm for problem ($\star$)

Input: $\{\mathbf{y}_r\}_{r=1}^{R}$, γ;
Initialize: $\left\{\mathbf{x}_k^{(0)}\right\}_{k=1}^{S}$, $\left\{\mathbf{h}_{rk}^{(0)}\right\}_{k,r=1}^{S,R}$;
repeat
  update $\{\mathbf{x}_k\}_{k=1}^{S}$ in parallel by equation (10);
  update $\{\mathbf{h}_{rk}\}_{k,r=1}^{S,R}$ in parallel by equation (14);
until some convergence criterion is satisfied
Output: $\{\mathbf{x}_k\}_{k=1}^{S}$, $\{\mathbf{h}_{rk}\}_{k,r=1}^{S,R}$.

C. Complexity Analysis

We analyze the per-iteration computational complexity of Algorithm 1. In each iteration, the computational cost of updating signals comes from matrix multiplication in $\mathbf{m}$, with time complexity being $\mathcal{O}\left(MNRS^2 \log M\right)$; and the computational cost of updating filters comes from matrix multiplication in $\mathbf{N}$, being $\mathcal{O}\left(MNRS^2 \log M\right)$. Time complexity of $\mathcal{O}\left(MN \log M\right)$ is due to the use of fast Fourier transform in some matrix multiplications.

V. Interpreting BMM as Alternating Proximal Gradient Descent

In developing BMM, the surrogates (8) and (12) are the isotropic quadratic approximation of the objectives in (7) and (11), respectively. This provides BMM an intriguing parallel to the alternating proximal gradient descent method. Denote the fitting-error term in the objective ($\star$) as f. The gradients of f with respect to $\mathbf{x}_k$ and $\mathbf{h}_{rk}$ are $\nabla_{\mathbf{x}_k} f$ and $\nabla_{\mathbf{h}_{rk}} f$, respectively.

In updating the signals, by plugging the expression of $\mathbf{m}_k$ into (10), we can verify that the update rule of $\mathbf{x}_k$ is the proximal gradient descent with stepsize C^{-1}, i.e.

$$\begin{aligned}\mathbf{x}_k^{+} &= \operatorname{prox}_{C^{-1}\gamma\|\cdot\|_1}\left(C^{-1}\mathbf{m}_k\right)\\ &= \operatorname{prox}_{C^{-1}\gamma\|\cdot\|_1}\left(\underline{\mathbf{x}}_k - C^{-1}\nabla_{\mathbf{x}_k} f\right),\end{aligned} \tag{15}$$

where $\operatorname{prox}_{\lambda g(\cdot)}(\mathbf{z}) = \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \lambda g(\mathbf{x})$. Likewise, in the filter-updating block, we have

$$\begin{aligned}\mathbf{h}_{rk}^{+} &= \frac{\mathbf{n}_{rk}}{\|\mathbf{n}_{rk}\|}\\ &= \operatorname{prox}_{\mathbb{I}(\|\cdot\|=1)}\left(D^{-1}\mathbf{n}_{rk}\right)\\ &= \operatorname{prox}_{\mathbb{I}(\|\cdot\|=1)}\left(\underline{\mathbf{h}}_{rk} - D^{-1}\nabla_{\mathbf{h}_{rk}} f\right).\end{aligned} \tag{16}$$

Therefore, the two BMM updates can be viewed as proximal gradient descent steps for both $\mathbf{x}_k$'s and $\mathbf{h}_{rk}$'s.

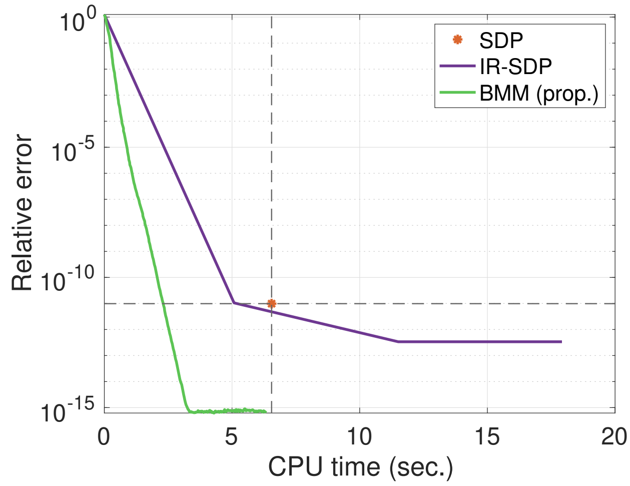

Fig. 2: Comparison of CPU time.

VI. Numerical Experiments

In this section, we illustrate the recovery performance of the proposed algorithm and compare it with existing methods. Suppose we have $S = 10$ devices in which only 2 send messages to a base station that has $R = 2$ antennas. We draw nonzero elements of the ground truth $\{\mathbf{x}_k^{\star}\}_{k=1}^{S}$ and $\{\mathbf{h}_{rk}^{\star}\}_{k,r=1}^{S,R}$ independently from the standard complex Gaussian distribution and set $\mathbf{h}_{rk}^{\star}$'s to have unit length. The encoding matrices $\mathbf{A}_k$'s are chosen based on the design in Section II. 20 Monte Carlo experiments are performed and the average results are presented.

First we conduct experiments in the noiseless case to compare the convergence speed of BMM with existing methods, including SDP [9] and IR-SDP [12], where the first is the convex method and the second employs a nonconvex low-rank inducing function. We set $M = 128$ and $N = L = 8$. The relative error (RE) between the estimation and the ground truth

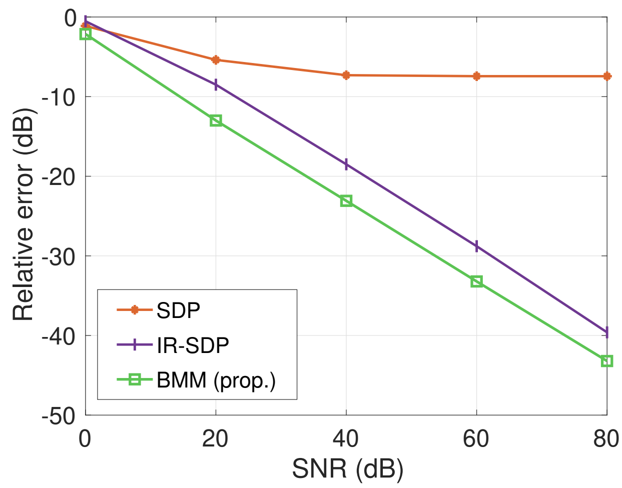

Fig. 3: Performance under different SNR.

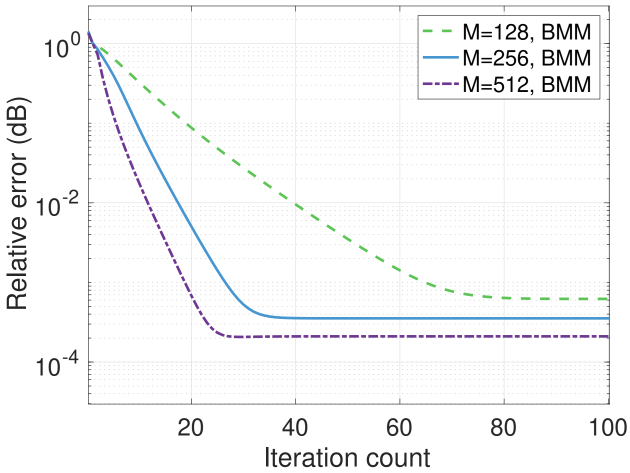

Fig. 4: Effectiveness of the proposed BMM.

is defined as

$$\mathrm{RE} = \frac{\sum_{r=1}^{R}\sum_{k=1}^{S}\left\|\hat{\mathbf{h}}_{rk}\hat{\mathbf{x}}_k^{\mathsf{H}} - \mathbf{h}_{rk}^{\star}\mathbf{x}_k^{\star\mathsf{H}}\right\|}{\sum_{r=1}^{R}\sum_{k=1}^{S}\left\|\mathbf{h}_{rk}^{\star}\mathbf{x}_k^{\star\mathsf{H}}\right\|}.$$

RE of each method against CPU time is presented in Fig. 2. It shows that BMM converges faster than SDP and IR-SDP. Then we compare the recovery performance of the algorithms in a noisy scenario. We set $M = 128$, $N = L = 16$, and choose the noise to be complex white Gaussian. The signal-to-noise ratio (SNR) is defined as

$$\mathrm{SNR} = 10\log_{10}\frac{\sum_{r=1}^{R}\sum_{k=1}^{S}\left\|\mathbf{h}_{rk}^{\star}\mathbf{x}_k^{\star\mathsf{H}}\right\|}{\sum_{r=1}^{R}\left\|\mathbf{e}_r\right\|^2}.$$

Fig. 3 shows that BMM obtains the best recovery performance. Moreover, we perform BMM with different sample sizes to verify its effectiveness. For $M \in \{128, 256, 512\}$, we set $N = L = 16$ and SNR $= 60$. It is interesting to observe that BMM converges linearly despite of the problem size, as shown in Fig. 4. Finally, to see the scalability of these methods, we implement them under different settings and compare their CPU time costs when they converge. As displayed in Table I, BMM takes far less time than existing methods and hence is scalable to large-scale problems.

VII. Conclusions

In this paper, we have proposed a BMM algorithm to jointly detect active devices and recover signals in MIMO communications, without prior knowledge of channel information.

TABLE I: CPU time (sec.) comparison in the noiseless case.

M	$N = L$	IR − SDP	SDP	BMM (prop.)
256	8	28.05	8.93	**2.09**
	16	261.39	89.57	**4.52**
512	16	693.55	188.10	**3.49**
	32	11006.13	4370.12	**13.72**

Based on the matrix factorization structure, the signals and the filters have been updated in an alternating way with low-cost closed-form solutions. Numerical results have indicated that BMM not only outperforms both SDP and IR-SDP methods in terms of recovery accuracy but also reduces the computational time significantly. Theoretical analyses of the convergence rate and performance guarantees in MIMO systems have not yet been provided, which suggests a potential future direction for our research.

References

[1] E. Bastug, M. Bennis, M. Medard, and M. Debbah, "Toward interconnected virtual reality: Opportunities, challenges, and enablers," *IEEE Comm. Mag.*, vol. 55, no. 6, pp. 110–117, 2017.

[2] K. S. Kim, D. K. Kim, C. Chae, S. Choi, Y. Ko, J. Kim, Y. Lim, M. Yang, S. Kim, B. Lim, K. Lee, and K. L. Ryu, "Ultrareliable and low-latency communication techniques for tactile internet services," *Proc. of the IEEE*, vol. 107, no. 2, pp. 376–393, 2019.

[3] L. Liu and W. Yu, "Massive connectivity with massive MIMO-Part I: Device activity detection and channel estimation," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2933–2946, 2018.

[4] J. Ahn, B. Shim, and K. B. Lee, "Ep-based joint active user detection and channel estimation for massive machine-type communications," *IEEE Trans. Comm.*, vol. 67, no. 7, pp. 5178–5189, 2019.

[5] L. Liu and W. Yu, "Massive connectivity with massive MIMO-Part II: Achievable rate characterization," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2947–2959, 2018.

[6] M. Chen and Z. Zhao, "Joint blind deconvolution and demixing of sparse signals via factorization and nonconvex optimization," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2024, pp. 9176–9180.

[7] A. Ahmed, B. Recht, and J. Romberg, "Blind deconvolution using convex programming," *IEEE Trans. Inf. Theory*, vol. 60, no. 3, pp. 1711–1732, 2014.

[8] A. Ahmed, "A convex approach to blind MIMO communications," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 812–815, 2018.

[9] S. Oymak, A. Jalali, M. Fazel, Y. C. Eldar, and B. Hassibi, "Simultaneously structured models with application to sparse and low-rank matrices," *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2886–2908, 2015.

[10] M. Fazel, H. Hindi, and S. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *Proc. Amer. Control Conf.*, vol. 6, 2001, pp. 4734–4739.

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

[12] Y. Shi, J. Dong, and J. Zhang, *Low-overhead Communications in IoT Networks: Structured Signal Processing Approaches*, English. Springer, 2020, ISBN: 978-981-15-3869-8.

[13] J. Y. Gotoh, A. Takeda, and K. Tono, "DC formulations and algorithms for sparse optimization problems," *Math. Program.*, vol. 169, no. 1, pp. 141–176, 2018.

[14] M. Razaviyayn, M. Hong, and Z. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 1126–1153, 2013.

[15] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 794–816, 2017.

[16] A. Nedić, J. Pang, G. Scutari, and Y. Sun, *Parallel and distributed successive convex approximation methods for big-data optimization*. Springer, 2018, vol. 2224, pp. 141–308.

[17] S. Ling and T. Strohmer, "Blind deconvolution meets blind demixing: Algorithms and performance bounds," *IEEE Trans. Inf. Theory*, vol. 63, no. 7, pp. 4497–4520, 2017.

[18] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[19] N. Parikh and S. P. Boyd, "Proximal algorithms," *Foundations Trends Optim.*, vol. 1, no. 3, pp. 127–239, 2014.