An Efficient Two-Stage SPARC Decoder for Massive MIMO Unsourced Random Access

Juntao You, Member, IEEE, Wenjie Wang, Member, IEEE, Shansuo Liang, Member, IEEE, Wei Han, Member, IEEE, and Bo Bai, Senior Member, IEEE

Abstract—In this paper, we study a concatenate coding scheme based on sparse regression code (SPARC) and tree code for unsourced random access in massive multiple-input and multiple-output systems. Our focus is concentrated on efficient decoding for the inner SPARC with practical concerns. A two-stage method is proposed to achieve near-optimal performance while maintaining low computational complexity. Specifically, a one-step thresholding-based algorithm is first used for reducing large dimensions of the SPARC decoding, after which a relaxed maximum-likelihood estimator is employed for refinement. Adequate simulation results are provided to validate the near-optimal performance and the low computational complexity. Besides, for covariance-based sparse recovery method, theoretical analysis is given to reveal the upper bound of the number of active users supported when convex relaxation is considered, and the probability of successful dimension reduction by the one-step thresholding-based algorithm.

Index Terms—Unsourced random access, massive MIMO, internet of things, SPARC, approximate message passing.

I. INTRODUCTION

M ASSIVE machine type communication (mMTC) is one of the primary advancements in the internet of things (IoT) applications [1], [2], [3]. In a typical mMTC scenario, a massive amount of potential user equipment (UE) sends sporadic data to a base station (BS) and the BS needs to decode the message from active users [4], [5], [6], [7]. In general, the UE in mMTC is a low-cost and battery-powered sensing device and has limited computational resources. Within a certain time slot, only a portion of potential UE would be active and send short packets (around $10^2$ bits/packet) to the BS and the UE in mMTC is usually latency insensitive. Overall, in mMTC, the BS could support a massive amount of UE with decoding algorithms of slightly higher computational complexity.

Most cellular standards, such as 3G, 4G-LTE and 5G-NR, utilize grant-based schemes for transmission [2], [3]. In the grant-based schemes, each active user is associated with a unique preamble to perform random access procedures, where

A user could identify itself and ask for transmission resources. The BS proactively schedules transmission resources for each identified user. However, the sporadic communication patterns and the massive amount of potential users in mMTC cause huge scheduling overhead, rendering such grant-based schemes inefficient and impractical. Therefore, grant-free based protocols have attracted significant attention recently, where UE sends data directly to the BS without approval. With carefully designed collision resolution mechanisms, grant-free schemes require small scheduling overhead and are highly efficient, which consequently seems favorable for mMTC [2], [8].

Recently, a new grant-free random access paradigm called “unsourced random access” (URA) was proposed for mMTC [4], [7]. In URA, $K_{total}$ potential users employ the same codebook and transmit messages sporadically without revealing their identities. The receiver aims to recover a list of permuted messages without knowing the identities of $K \ll K_{total}$ active users [5]. Particularly, we consider an uplink multi-user grant-free system over a block-fading wireless communication channel, where the channel coefficients remain constant over coherence blocks of $D$ channel uses, and change randomly from block to block according to a stationary ergodic process [9].

A. Prior Work

Various schemes have been proposed for URA in AWGN channels, such as $T$-fold ALOHA [10], coded compressive sensing [11], [12] and sparse regression code (SPARC) [5]. Considering the channel fading in a wireless system, the Reed-Muller-based scheme [13] is proposed for URA with a single antenna at BS. However, it does not apply to the multiple-input multiple-output (MIMO) case, where BS is equipped with multiple antennas and the multiple inputs come from the data streams of different users [1], [6]. While the $T$-fold ALOHA scheme [14], [15] and the differential coding scheme [7] can be applied to the MIMO case, both of these schemes utilize clustering and successive interference cancellation (SIC) for data decoding, which incurs high computational complexity and delay when active user number $K$ is large. A pilot-based coherent scheme was proposed in [1] and [16] for URA in the MIMO scenario, where orthogonal or non-orthogonal pilots are used for active detection and channel estimation. Then,
the maximum-ratio combining method using the estimated channels is employed for MIMO detection. It requires that the estimated channels vary slowly from the beginning of the pilot transmission to the end of the data transmission. That is, this pilot-based scheme requires coherence blocks of large channel uses (around $10^3$), which is far from practical in URA.

Alternatively, a concatenated coding scheme was proposed in [5], and [6] for URA to use SPARC and tree code as inner and outer codes, respectively. The inner SPARC uses a common codebook matrix of size $D \times N$ for all the potential users, where up to $\log_2 N$ information bits can be transmitted. To reduce the storage and computational complexity for SPARC involving large $N$, tree code is used as outer code to enable dividing the original information bits into several sections that are separately encoded by SPARC with much smaller $N$. Other choices for the outer code include cyclic redundancy check and Reed-Solomon code [17]. At the receiver, non-coherent detections are used for decoding SPARC section by section and the outer tree decoder stitches different sections of messages together to produce original information messages.

For the inner SPARC, a series of recent works [18], [19] have formulated it as a multiple-measurement-vector (MMV) problem. AMP-type algorithms are applied as a popular approach for such an MMV problem [18], [20]. However, the compressive-sensing-based approaches such as AMP [18] and Orthogonal AMP (OAMP) [21] perform poorly when the number of active users $K > D$ [5], [6]. The above limitation means that the dimensions of the codebook matrix $D$ must scale linearly with $K$, which could be infeasible in practice. Alternatively, covariance-based non-coherent estimators are proposed for the SPARC decoding, such as (relaxed) maximum-likelihood (ML) and non-negative least squares (NNLS), which are shown to have better empirical performance than AMP at the cost of higher computational complexity [6], [22].

### B. Contributions

The paper is focused on the SPARC decoding with practical concerns. Essentially, existing AMP [18], OAMP [21] and covariance-based non-coherent estimators [22] require the channel to stay static over $D$ channel uses during the transmission of the codeword. However, due to physical environment constraints, the coherence block of the uplink channel is usually limited around hundreds of channel uses in most existing wireless cellular systems [9], [15], where both AMP and OAMP deteriorate significantly [5]. Furthermore, the number of active users supported by compressive-sensing-based approaches, e.g., AMP [18], OAMP [21], ISTA [23], FISTA [24], ADMM [25], is bounded by $O(D^2)$ [22], [26]. On the other hand, the covariance-based approaches perform remarkably well for small $D$ and the number of supported users can achieve $O(D^2)$ [22]. The main issue for covariance-based approaches is their high computational complexity.

In this paper, we concentrate on the covariance-based approaches and propose a two-stage method called accelerated maximum-likelihood which combines a one-step thresholding-based algorithm for dimension reduction and ML for refinement. Due to the dimension reduced in the first stage, the complexity of ML is significantly reduced while the overall performance still being maintained. Adequate numerical results are provided to support such claims, including the link-level simulations over the existing cellular systems. On the other hand, theoretical analyses are provided for both the covariance-based sparse recovery model and the thresholding algorithm. For the former, the optimal upper bound for the number of active users $K$ is provided when convex relaxation is considered. For the latter, the one-step thresholding-based algorithm is theoretically guaranteed to contain all the active users with high probability when the number of BS antennas $M$ and channel uses $D$ are reasonably large.

The structure of this paper is as follows: The standard concatenated coding scheme of SPARC and tree code [6] is introduced in Section II while our proposed algorithm is introduced in Section III. Theoretical analyses for both the covariance-based sparse recovery model and the thresholding algorithm are provided in Section IV. In Section V, simulations are provided to demonstrate the performance of the proposed algorithm. Finally, this work is concluded in Section VI.

### C. Notations

In the paper, $\mathbb{R}$ denotes the set of real numbers, $\mathbb{C}$ denotes the set of complex numbers, and $\mathbb{B}$ denotes the set of binary numbers. For any $a \in \mathbb{C}$, we denote $\bar{a}$ the conjugate of $a$. For any vector $x \in \mathbb{R}^N$ and any matrix $A \in \mathbb{R}^{D \times N}$, $x^T$ and $A^T$ are their transpose, respectively. For $x \in \mathbb{C}^N$ and $A \in \mathbb{C}^{D \times N}$, $A^*$ and $A^T$ are their conjugate (Hermitian) transpose, respectively. $A(i,:)$, $A(:,j)$ are the $i$-th row and $j$-th column of $A$ respectively. The notation $\|x\|_0$ stands for the number of nonzero entries of $x$, and $\|x\|_1$, $\|x\|_2$ are the standard $\ell_1$-norm, $\ell_2$-norm of $x$ respectively. $\|A\|_2$ and $\|A\|_F$ denote the spectral norm and Frobenius norm of matrix $A$ respectively. For $a \in \mathbb{R}$, $(a)_+ = \max(a, 0)$. For a positive integer $n$, the notation $[n]$ represents $\{1, 2, \ldots, n\}$. For $x \in \mathbb{R}^N$, $x_{\min}$, $x_{\max}$ are the smallest and largest components in magnitude of $x$, respectively. For $\gamma \in \mathbb{R}^N_+$, it means that $\gamma \in \mathbb{R}^N$ and $\gamma_i \geq 0$, $\forall i \in [N]$. For any $\gamma \in \mathbb{R}^N$, the notation $\text{diag}(\gamma)$ means the $N \times N$ diagonal matrix whose diagonal entries are given by $\{\gamma_i\}_{i=1}^N$. For a set $S$, $|S|$ denotes the number of elements in $S$. For any positive integer $s$ and any vector $x \in \mathbb{C}^N$, $H_x$ denotes the hard thresholding operator that keeps only the largest $s$ components (in magnitude) of $x$ and set the others to be zero. For matrix $A \in \mathbb{C}^{D \times N}$, $\text{vec}(A)$ denotes the $DN \times 1$ vector obtained by stacking the columns of $A$.}

### II. Preliminaries

In the standard concatenated coding scheme for the URA [5], [6] in Fig. 1, a frame of $D_{\text{total}}$ channel uses is
shared by $K_{\text{total}}$ potential users and only $K \ll N \leq K_{\text{total}}$ users are active\(^4\). Each active user transmits an information message of $B$ bits, i.e., $x_{l,k} \in \mathbb{B}^B$. It should be highlighted that channel state information, i.e., $g_{l,k}$ and $h_{l,k}$ in Fig. 1, is available at neither users nor BS.

### A. Encoding

The information message is first encoded by an outer tree code [11] as shown in Fig. 1. Specifically, the $\alpha$-th user's information message $x_\alpha$ is split as $x_{1,\alpha}, x_{2,\alpha}, \ldots, x_{L,\alpha}$. The $L$ sections are of length $b_1, b_2, \ldots, b_L$, where $\sum_{l=1}^{L} b_l = B$ and $b_1 = J$ and $b_l < J$ for $l = 2, \ldots, L$. Each section is augmented to size $J$ by appending $p_l = J - b_l$ parity bits. The coding rate of tree code is $R_{\text{tree}} = \frac{B}{2J}$. For section $l = 2, 3, \ldots, L$, $p_l$ parity bits are generated out of the previous $l-1$ sections of information bits using a random binary matrix $G_l \in \mathbb{B}^{b_l \times m_l}$, where $m_l = \sum_{l=1}^{l-1} b_l$. The random binary matrices $\{G_l\}_{l=2}^L$ are identical for all users. A total number of $P = \sum_{l=2}^{L} p_l$ parity bits are used to link $L$ sections of information bits together to produce the original information message at the receiver. The $L$ coded messages of user $\alpha$ is denoted as $\pi_1, \pi_2, \ldots, \pi_{L,\alpha}$, where $\pi_{l,\alpha} \in \{2^J\}$ is a decimal representation of a $J$-bit binary message.

For the inner code, each user will apply SPARC individually to encode each $\pi_{l,\alpha}$ into a length $D = D_{\text{total}}/L$ codeword separately [6]. Let $A = [a_1, a_2, \ldots, a_N] \in \mathbb{C}^{D \times N}$ with $N = 2^J$ be the codebook matrix that is shared by all potential users. The codebook matrix $A$ is assumed to be a column-normalized random matrix, e.g., the matrix whose columns are i.i.d. random variables drawn uniformly from the sphere, Bernoulli random matrix, sub-sampled Fourier matrix, etc. The sub-sampled Fourier matrix is preferred from the computational point of view. It has been empirically shown to have no performance loss in terms of reconstruction error when compared with the random i.i.d. Gaussian ones in the context of compressive sensing [27], which is further confirmed by the replica analysis [28], [29]. Each user $\alpha$ chooses one column of the codebook based on its own $J$-bit coded message $\pi_{l,\alpha}$ as its codeword $a_{l,\alpha} \in \mathbb{C}^D$. That is, the index $\pi_{l,\alpha}$ of the selected column of $A$ represents the $J$-bit message. The coding rate of SPARC is $R_{\text{SPARC}} = \frac{B}{D}$. Concatenated with tree code, the information rate of each user is given by $R = R_{\text{SPARC}} \times R_{\text{tree}} = \frac{B}{D_{\text{total}}}$ and the sum rate of $K$ active users is given by $R_{\text{total}} = \frac{B}{D_{\text{total}}}$.

### B. Decoding

We denote the set of active users as $\mathcal{K} \subseteq [K_{\text{total}}]$ and $|\mathcal{K}| = K$. Assuming that BS is equipped with $M$ antennas, the received signal $Y_l \in \mathbb{C}^{D \times M}$ is given by

$$Y_l = \sum_{\alpha \in \mathcal{K}} \sqrt{g_{l,\alpha}} a_{l,\alpha}^T h_{l,\alpha} + Z_l, \quad l = 1, 2, \ldots, L,$$  \hspace{1cm} (1)

where $h_{l,\alpha} \in \mathbb{C}^M$ is the user channel vector of small-scale fading coefficients, $g_{l,\alpha} \in \mathbb{R}_+$ is the large-scale fading coefficient (LSFC) and $Z_l \in \mathbb{C}^{D \times M}$ is the additive white Gaussian noise (AWGN).

Furthermore, for $K$ active users in section $l$, collision will occur if different active users happen to choose the same codeword. Taking collision into consideration, we define the index set $Q_k := \{\alpha | \pi_{l,\alpha} = k, \alpha \in \mathcal{K}\}, \ k \in [N]$. Considering Rayleigh fading channel, the user channel vector $h_{l,\alpha} \in \mathbb{C}^M$ follows $CN(0, I_M)$. We then define a vector $\gamma_l = [\gamma_{l,1}, \gamma_{l,2}, \ldots, \gamma_{l,N}]$ and a sequence of channel vectors $\{h_l\}_{k=1}^N \in \mathbb{C}^M$, which satisfy $\sqrt{\gamma_{l,k}} h_{l,k} := \sum_{\alpha \in Q_k} \sqrt{g_{l,\alpha}} h_{l,\alpha}$. Note that $\gamma_{l,k}$ is re-scaled so that $h_{l,k}$ also follows $CN(0, I_M)$. If $Q_k$ is empty, $\gamma_{l,k} = 0$ and $h_{l,k}$ are arbitrary samples from $CN(0, I_M)$. The channel matrix can then be denoted as $H_l = [h_{l,1}, h_{l,2}, \ldots, h_{l,N}]^T \in \mathbb{C}^{N \times M}$.

Since the inner encoder and decoder are identical for each section, with a slight abuse of notation, the subscription $l$ in (1) and the corresponding matrices/vectors are omitted for simplicity. The received signal for each section can be written in matrix form as

$$Y = A\sqrt{\Gamma} H + Z,$$  \hspace{1cm} (2)

\(^4\)The inner SPARC can support an arbitrary number of potential users such that the exact $K_{\text{total}}$ is generally irrelevant.
where the entries of unknown \( \mathbf{H} \) and \( \mathbf{Z} \) are considered to be mutually independent and distributed as \( \mathcal{CN}(0, 1) \) and \( \mathcal{CN}(0, \sigma^2) \), respectively, and \( \mathbf{F}_t \in \mathbb{C}^{N \times N} \) is a diagonal matrix whose diagonal entries are given by \( \gamma^t \). Recall that at each time slot, the number of active users \( K \) is much smaller than the codebook size, i.e., \( K \ll N \). That is, we have
\[
\|\gamma^t\|_0 \leq K, \quad K \ll N.
\]
The set of the coded messages of \( K \) users in each section is denoted by \( \text{supp}(\gamma^t) \), which is the set of the index of the non-zero entries of \( \gamma^t \). The problem we study is to recover the transmitted signal, which can be determined by \( \text{supp}(\gamma^s) \), at the BS with unknown \( \tilde{g}_{i,t}^s \) and \( \tilde{h}_{i,t}^s \).

The ML estimator for (2) is to obtain the estimate \( \hat{\gamma} \) from the sample covariance matrix \( \hat{\Sigma}_y \).

The ML algorithm will output an estimate of \( \gamma^s \) as \( \hat{\gamma} \), where the \( k \)-th entry \( \hat{\gamma}_k \) denotes the possibility of \( a_k \) being transmitted by an active user and \( \hat{\gamma}_k = 0 \) if this codeword is not selected by any active user.

After performing the ML algorithm, the tree decoding step is introduced as shown in Fig. 1. While active user number \( K \) is unknown at the BS, it is reasonable to assume that the maximal number of active user \( T \) is known \([10],[15]\). It should be noted that \( T \) is not required in the inner SPARC decoding, it is only used in the input of outer tree decoding. Nevertheless, the largest \( T \) components in \( \hat{\gamma} \) of section \( l \) are selected and their indices are written \( \mathcal{L}(Y_l) = \{ \tilde{h}_{i,1}, \tilde{h}_{i,2}, \ldots, \tilde{h}_{i,T} \} \). There are a total of \( L \) sets of messages, i.e., \( \{ \mathcal{L}(Y_l) \}_{l=1}^L \), each of which contains \( T \) messages and each message conveys \( J \) coded bits. For each message, the tree decoding aims to link its \( L \) sections together with parity bits and remove falsely detected coded messages simultaneously. In this paper, our focus is concentrated on the inner decoder while details for tree decoding can be found in \([11], \text{Section IV-C}\).

III. THE PROPOSED TWO-STAGE ALGORITHM

In this section, a covariance-based approach is proposed to decode SPARC in two stages: a one-step thresholding-based algorithm for rough support recovery and an ML estimator for refinement. First, Eq. (2) is reformulated into a sparse vector recovery problem, where the thresholding-based algorithms are introduced. Then, we give a brief review of the ML estimator which numerically performs well in the SPARC decoding. Last, the two-stage method is introduced to reduce the overall computational cost while maintaining similar performance as ML.

A. Recovery of Sparse Vectors From Covariance Matrix

Define the columns of channel matrix as \( \tilde{\mathbf{h}}_j := \mathbf{H}(:,j) \) and columns of additive white Gaussian noise matrix as \( \tilde{\mathbf{z}}_j := \mathbf{Z}(:,j) \). The columns of \( \mathbf{Y} \) in (2) can be written as \( \mathbf{Y}(:,j) = A \mathbf{F}_t^{1/2} \tilde{\mathbf{h}}_j + \tilde{\mathbf{z}}_j \). Recall that the entries of \( \mathbf{H} \) and \( \mathbf{Z} \) are considered to be mutually independent and distributed as \( \mathcal{CN}(0,1) \) and \( \mathcal{CN}(0,\sigma^2) \), respectively. Then, the columns of \( \mathbf{Y} \) are independent samples from a multivariate complex Gaussian distribution as \( \mathbf{Y}(:,j) \sim \mathcal{CN}(0,\Sigma_y) \), where the covariance matrix \( \Sigma_y \) can be calculated as
\[
\Sigma_y = \mathbb{E}[\mathbf{Y}(:,j)\mathbf{Y}(:,j)^*] = A \sqrt{\mathbf{F}_t^{1/2}} \mathbf{h}_j^* \sqrt{\mathbf{F}_t^{1/2}} \mathbf{A}^* + \sigma^2 \mathbf{I}_D
\]
\[
= A \mathbf{F}_t^{1/2} \mathbf{A}^* + \sigma^2 \mathbf{I}_D.
\]

Note that the sample covariance matrix \( \hat{\Sigma}_y = \frac{1}{M} \mathbf{Y} \mathbf{Y}^* \rightarrow \Sigma_y \) if \( M \rightarrow \infty \). Thus for sufficiently large \( M \), \( \gamma^s \) can be approximately estimated from the sample covariance matrix \( \hat{\Sigma}_y \) by solving the following problem: find \( \gamma^s \in \mathbb{R}_+^N \) that satisfies
\[
\hat{\Sigma}_y = A \text{diag}(\gamma^s) \mathbf{A}^* + \sigma^2 \mathbf{I}_D + \Delta, \quad \text{s.t.} \quad \|\gamma^s\|_0 \leq K,
\]
where \( \Delta = \hat{\Sigma}_y - \Sigma_y \) is the unknown error matrix between the sample and true covariance matrix. To bound \( \Delta \), it has been shown in \([6], \text{Theorem 11}\) that
\[
P\left( \|\Delta\|_F \leq c_\varepsilon D \left( \|\gamma^s\|_1 + \sigma^2 \sqrt{M} \right) \right) \geq 1 - \varepsilon
\]
for \( \|a_k\|_0^2 = D \) and sufficiently large \( M \) (see also Lemma 5).

Here, \( c_\varepsilon = \sqrt{c \log \left( \frac{cM}{\varepsilon} \right)} \), where \( \varepsilon \in (0,1) \) is arbitrary and \( c \) is an universal constant. It then implies that \( \|\Delta\|_F \) is small when \( M \) is sufficiently large. In view of this, the problem (5) can be regarded as a robust sparse vector recovery problem. Hence, it is natural to consider solving the following \( \ell_0 \)-minimization problem
\[
\min_{\gamma \in \mathbb{R}_+^N} \|\gamma\|_0, \quad \text{s.t.} \quad \|\Sigma(\gamma) - \hat{\Sigma}_y\|_F \leq \eta,
\]
where \( \Sigma(\gamma) := A \text{diag}(\gamma) \mathbf{A}^* + \sigma^2 \mathbf{I}_D \), and \( \eta := c_\varepsilon D (\|\gamma^s\|_1 + \sigma^2 \sqrt{M}) \) according to (6). If the linear equation in (5) is underdetermined, it is widely known that the \( \ell_0 \)-minimization problem (7) is NP-hard in general (see \([30], \text{Theorem 2.17}\)). Thus, one may consider its convex relaxation in form of \( \ell_1 \)-minimization as
\[
\min_{\gamma \in \mathbb{R}^N} \|\gamma\|_1, \quad \text{s.t.} \quad \|\Sigma(\gamma) - \hat{\Sigma}_y\|_F \leq \eta,
\]
where the non-negativity in \( \gamma \) is also relaxed. The recovery guarantees of the basis pursuit approach (8) can be found in Section IV-A, showing that (8) has exact recovery of \( \gamma^s \) with near-optimal scaling law in \( K \) and \( D \). Problem (8) is convex and many algorithms can be applied, e.g., ISTA \([23]\), FISTA \([24]\), ADMM \([25]\). Though convergence of such algorithms is well studied, there is few results that (8) can be exactly solved in a (very few) fixed number of iterations. In this work, we consider the non-convex iterative hard thresholding (IHT) algorithm in the following subsections. Theoretically, we show that the IHT algorithm can be guaranteed to have exact support recovery in just one step — meaning that the algorithm can be designed to be non-iterative and super efficient.
B. One-Step IHT Algorithm

In this sub-section, we apply a first-order gradient-type non-convex algorithm for (5). To begin with, we define a matrix \( \mathcal{A} \in \mathbb{C}^{D^2 \times N} \) where its \( k \)-th column is given by
\[
\mathcal{A}(i, k) := \text{vec}(a_k a_k^*), \quad k \in [N].
\]
(9)
Relaxing the non-negativity in \( \gamma^2 \), the problem (5) can be reformulated as
\[
u = \mathcal{A} \gamma + \delta, \quad \text{s.t.} \quad \|\gamma\|_0 \leq K,
\]
(10)
where \( \nu := \text{vec}(\Sigma y - \sigma^2 I_D) \) and \( \delta := \text{vec}(\Delta) \). Consider the standard least squares fitting for (10). We then have the following minimization problem
\[
\min_{\|\gamma\|_0 \leq K} f_I(\gamma), \quad \text{where} \quad f_I(\gamma) := \frac{1}{2D^2} \|\nu - \mathcal{A} \gamma\|^2_2.
\]
(11)
Therefore, we can directly apply the projected gradient descent (PGD) algorithm for (11). When the hard thresholding operator is employed for projection, the PGD is equivalent to the IHT algorithm [31]. The update at \( t \)-th iteration of IHT is given by
\[
\gamma_{t+1} = \mathcal{H}_K(\gamma_t + \alpha \nabla f_I(\gamma_t))
\]
\[
= \mathcal{H}_K(\gamma_t + \frac{\alpha}{\mathcal{A}^*} (\nu - \mathcal{A} \gamma_t)), \quad t \geq 0,
\]
(12)
where \( \alpha > 0 \) is the step size and the hard thresholding operator \( \mathcal{H}_K \) is to project the update into the space \( \{\gamma \in \mathbb{R}^N : \|\gamma\|_0 \leq K\} \). For different codebooks \( \mathcal{A} \), the linear convergence guarantees of IHT in (12) are provided in Section IV-A.

Notice the algorithm relies on a proper choice of step size and iteration number (see Theorem 2). In fact, we are aiming at support recovery in this work, and thus the recovery of \( \gamma^2 \) is not essentially required (see Fig. 1). Therefore, we consider a much more simple and efficient non-iterative method – the one-step IHT algorithm, see Algorithm 1.

Algorithm 1 One-Step IHT Algorithm

Initialize: \( \gamma_0 = 0 \).
1. \( \gamma_1 = \mathcal{H}_K(\gamma_0 + \frac{1}{\mathcal{A}^*} \nu - \mathcal{A} \gamma_0)) \)
2. \( S_1 = \text{supp}(\gamma_1) \)
Output: The resulting estimate \( S_1 \).

Furthermore, for one step of IHT, we show that w.h.p. it is guaranteed to have exact support recovery (i.e. \( S_1 = \text{supp}(\gamma^2) \)) in Theorem 3, Section IV-B. To further improve the performance of the algorithm, we combine the one-step IHT with an ML estimator in the following sections.

C. The ML Estimation via Coordinate-Wise Optimization

In this subsection, we give a brief review of the ML estimation for the SPARC decoding and the corresponding iterative solver introduced in [22] and [6]. From (4), we consider the following negative log-likelihood cost function
\[
f(\gamma) := -\frac{1}{M} \log p(Y | \gamma) = -\frac{1}{M} \sum_{j=1}^{M} \log p(Y(\cdot, j) | \gamma)
\]
\[
\propto \log |\Sigma(\gamma)| + \text{tr}(\Sigma(\gamma)^{-1} \Sigma y).
\]
(13)

By relaxing the sparse constrained problem into a non-negative constrained problem, the relaxed ML estimator is given by
\[
\gamma^*_N = \arg \min_{\gamma \in \mathbb{R}^N} \left( \begin{array}{c} f(\gamma) \end{array} \right).
\]
(14)
A gradient-type algorithm based on the coordinate-wise optimization for solving (14) can be summarized as follows. For each coordinate \( k \in [N] \), define the scalar function
\[
f_k(d) = \log |\Sigma + da_k a_k^*| + \text{tr} \left( (\Sigma + da_k a_k^*)^{-1} \Sigma y \right),
\]
where \( \Sigma := \Sigma(\gamma) \) and \( \gamma \in \mathbb{R}^N \). By setting the derivative of \( f_k(d) \) as zero, i.e., \( f'_k(d) = 0 \), the solution is given by
\[
\hat{d}_k = \frac{a_k^* \Sigma^{-1} \Sigma y - a_k}{(a_k^* \Sigma^{-1} a_k)^2}.
\]
(15)
Then, one step of the coordinate-wise projected gradient descent update is given by: for all \( k \in [N] \), compute \( d_k = \max\{\hat{d}_k, -\gamma_k^*\} \) and update
\[
\gamma_k \leftarrow \gamma_k + d_k,
\]
(16a)
\[
\Sigma(\gamma) \leftarrow \Sigma(\gamma) + d_k a_k a_k^*,
\]
(16b)
where the \( \max\{\cdot\} \) is to ensure \( \gamma_k \in \mathbb{R}_+ \) (i.e., project \( \gamma_k \rightarrow \gamma_k^*_+ \)). Further, to avoid the expensive computational cost in calculating \( \Sigma^{-1} \), the rank-one update in (16b) can be replaced by
\[
\Sigma^{-1} \leftarrow \Sigma^{-1} - \frac{d_k \Sigma^{-1} a_k a_k^* \Sigma^{-1}}{1 + d_k a_k^* \Sigma^{-1} a_k},
\]
where the Sherman-Morrison update is applied. The iterative algorithm for solving (14) is then obtained by repeating the above coordinate-wise update.

Alternatively, if the standard least squares fitting for (5) is considered and the sparse constraint is relaxed to the non-negative constraint, the NNLS is then to solve the following convex problem [6], [22], [32]
\[
\gamma^*_N = \arg \min_{\gamma \in \mathbb{R}^N} \left( \begin{array}{c} \|\Sigma(\gamma) - \hat{\Sigma} y\|_F \end{array} \right).
\]
(17)
The NNLS estimator is guaranteed to identify the activity of \( \hat{O}(D^2) \) active users [6], [22]. Compared with NNLS estimator in (17), the ML estimator in (14) is non-convex, and thus the corresponding gradient algorithm falls short in theoretical guarantees. However, the ML estimator empirically outperforms the former significantly. In experiments, ML can successfully identify the activity of \( \hat{O}(D^2) \) active users. Therefore, the coordinate-wise iterative algorithm for ML in (14) is chosen for the refinement stage in our algorithm.

D. The Proposed Algorithm: A Two-Stage Method

In this section, we propose a two-stage method based on the thresholding-based algorithm and ML estimator. For the ML estimator, the existing gradient-type coordinate-wise minimization approaches are iterative algorithms on the entire index set \( \{1, 2, \ldots, N\} \). By utilizing the sparsity in \( \gamma^2 \), we introduce a fast coordinate-wise minimization method which reduces the computational cost dramatically in the first phase of the iteration. That is, we estimate a set \( S^0 \) that is
most likely to contain the support of $\gamma^2$ in the first stage of the algorithm. In the second stage, we adopt the coordinate minimization to the following problem

$$\hat{\gamma} = \mathop{\arg\min}_{\gamma \in \mathbb{R}_+^N} f(\gamma),$$  \hspace{1cm} (18)

where $f(\gamma)$ is defined in (13). To find out the indices that are most likely to be the support of $\gamma^2$ in the first stage, we define

$$\mathcal{Y}_k := \sum_{1 \leq i,j \leq D} \pi_{k,i} a_{k,i,j} \tilde{\Sigma}_y(i,j), \quad \forall k \in [N],$$  \hspace{1cm} (19)

where $\tilde{\Sigma}_y(i,j)$ is the $(i,j)$-th component of $\tilde{\Sigma}_y$. Recall that the channel and AWGN vectors are i.i.d. Gaussian. Thus we have the following proposition.

**Proposition 1:** Assume that $\{a_k\}_{k=1}^N$ are i.i.d. random variables drawn uniformly from the sphere of radius $\sqrt{D}$. For $\mathcal{Y}_k$ defined in (19), we have

$$E[\mathcal{Y}_k] = \begin{cases} \psi(D, \gamma^0, \sigma), & \forall k \notin S^0, \\ \psi(D, \gamma^0, \sigma) + \frac{D^3}{D + 1} \gamma^0, & \forall k \in S^0, \end{cases}$$

where $S^0 = \text{supp}(\gamma^0)$ and $\psi(D, \gamma^0, \sigma) := \sum_{\ell \in S^0} \frac{D^2}{D + 1} \gamma^0 + D\sigma^2$.

**Proof:** See Appendix B.

The Proposition 1 indicates that statistically the indices of the larger components of $\{\mathcal{Y}_k\}_{k=1}^N$ are the support of $\gamma^0$. Thus, it is natural to select larger components of $\{\mathcal{Y}_k\}_{k=1}^N$ as an initial guess of $S^0$, and we have the following Remark 1.

**Remark 1:** In fact, selecting larger components of $\{\mathcal{Y}_k\}_{k=1}^N$ is equivalent to the one-step IHT in Algorithm 1: Notice that the estimation of one-step IHT is given by

$$\gamma_t \leftarrow \gamma_{t-1} + d_t, \quad d_t = \frac{\gamma_{t-1} - \gamma_{t-2}}{\gamma_{t-1} - \gamma_{t-2}}.$$

**Proof:** See Appendix B.

In the next section, this algorithm will be analyzed thoroughly.

**IV. THEORETICAL RESULTS**

**A. Theoretical Guarantees of Sparse Recovery**

Let $\gamma$ be the solution to the convex optimization problem in (8). Theorem 1 below guarantees that the sparse vector recovery model in (8) admits a proper approximation to the underlying vector $\gamma^0$ provided that $M$ is sufficiently large.

**Theorem 1:** Let $\varepsilon > 0$ and assume that $\{a_k\}_{k=1}^N$ are i.i.d. random variables drawn uniformly from the sphere of radius $\sqrt{D}$. For $\gamma$ being the solution to (8), we have

$$\mathbb{P} \left( \|\gamma - \gamma^0\|_1 \leq c_1 \left( \|\gamma^0\|_1 + \sigma^2 \right) \sqrt{\log \left( \frac{(eD)^2}{\varepsilon} \right)} \right) \geq 1 - \varepsilon - c_2 \varepsilon^2,$$

provided that

$$K \leq c_3 D (D - 1) \log^2 (eN/D) \quad \text{and} \quad M \geq c \log \left( \frac{(eD)^2}{\varepsilon} \right),$$

where $c_1, c_2, c_3, c$ are universal positive constants.

**Proof:** See Appendix C.

A direct consequence of Theorem 1 is that

$$\|\gamma - \gamma^0\|_1 \leq c_1 \varepsilon$$

holds as long as $K \leq \mathcal{O} \left( D^2 / \log^2 \left( \frac{N}{D} \right) \right)$ and $M \geq \mathcal{O} \left( \frac{K \gamma_{\max} + \sigma^2}{\varepsilon} \right)^2 \log \left( \frac{(eD)^2}{\varepsilon} \right)$. Therefore, for $K \leq \mathcal{O} \left( D^2 / \log^2 \left( \frac{N}{D} \right) \right)$ and sufficiently large $M$, we will have arbitrary small $\varepsilon$ error between the recovered vector and the ground truth. Hence, it is possible for the covariance-based model to successfully detect at most $\mathcal{O}(D^2)$ active users. Further, the covariance-based sparse recovery model (8)
imposes a loose requirement on channel uses $D$ comparing with the traditional compressive sensing approach, which aims to recover $\sqrt{D}H$ from (2) [18], [20]. Given $K$ active users, Theorem 1 states that $D = O(\sqrt{K})$ is necessary. For comparison purposes, the traditional compressive sensing approach requires $D = O(K)$ [26].

When $M \to \infty$, we can see from (6) that $\|\delta\|_2 \to 0$, and (5) or (10) is reduced to find the $K$ sparse $\gamma^k$ from $u = A\gamma^k$. Therefore, to ensure that the linear system admits a unique solution (i.e., $\gamma^k$ be identifiable via the system), theoretically it requires each $2K$ columns of $A$ being linearly independent. Thus, the channel uses $D$ should satisfy $D^2 \geq 2K$ to recover every $K$ sparse vector. It implies that the bound in $D$ and $K$ given in Theorem 1 is near optimal.

Now we investigate the theoretical performance of the thresholding-based algorithms for (10). Due to its non-convexity, the convergence guarantee of IHT in (12) is not trivial. To ensure successful recovery, one common assumption is that the sensing matrix (e.g., $A$ in (10) satisfies a restricted isometry property (RIP) condition [33]. However, for general $\{a_k\}_{k=1}^{\infty}$ (e.g., i.i.d. Gaussian vectors), even if RIP holds for the rescaled version of $A = [a_1, a_2, \ldots, a_N]$, the matrix $A$ in (9) can not be shown to satisfy the RIP condition. Instead, we consider the mutual incoherence property (MIP) [34] of $A$. We theoretically show that the convergence of IHT can be guaranteed if the mutual coherence of the codebook matrix $A$ is small. Moreover, we will provide examples of the codebook matrix $A$ that satisfy the MIP condition.

Let

$$
\mu = \max_{k, \ell \in [N]} \frac{|\langle a_k, a_\ell \rangle|}{\|a_k\|_2 \|a_\ell\|_2},
$$

(21)

be the mutual coherence of $\{a_k\}_{k=1}^{N}$ (or the matrix $A$).

**Theorem 2:** Assume that $\|a_k\|_2 = \sqrt{D}$, $\forall k \in [N]$. Let $\mu$ be given in (21), $\eta$ be given in (7), and the sequence $\{\gamma_t\}_{t \geq 0}$ be defined by IHT in (12). For some constant size $\delta$, if $\{a_k\}_{k=1}^{N}$ satisfy the MIP condition $\mu \leq \frac{1}{3^{3/4} \sqrt{\lambda} - 1}$, then it holds

$$
\|\gamma_t - \gamma^k\|_2 \leq \epsilon \|\gamma_t - \gamma^k\|_2 + \epsilon \|\eta\|_2, \quad \forall t \geq 0,
$$

where $\epsilon \in (0, 1)$, $\zeta > 0$ are constants.

**Proof:** See Appendix D. \qed

Various choices of $\{a_k\}_{k=1}^{N}$ or $A$ are available to ensure small mutual coherence (see [30, Chapter 5]). Examples go as follows.

**Example 1:** Let $\{a_k\}_{k=1}^{N}$ be i.i.d. random variables drawn uniformly from the sphere of radius $\sqrt{D}$. Then, as long as $D > 8 \log(DN)$, we have

$$
P\left[ \mu \leq \sqrt{\frac{32 \log(DN)}{D}} \right] \geq 1 - \frac{c_0}{D^{1/2}},
$$

(22)

where $c_0$ is an universal constant.

**Example 2:** Let $A \in \mathbb{R}^{D \times N}$ be Bernoulli random matrix taking values $\pm 1$ with equal probability. Then, we have

$$
P\left[ \mu \leq 2 \sqrt{\frac{\log(N/\epsilon)}{D}} \right] \geq 1 - \epsilon^2.
$$

Example 1 is a direct consequence of (29) in Lemma 3 (see Appendix A, Example 2 is from [30, Exercise 9.1]. Together with Theorem 2, Eqs. (22) and (23) imply that it theoretically requires $D \geq O(K)$ to guarantee successful recovery of IHT algorithm.

**B. Recovery Guarantees of Algorithm 2**

The second stage of Algorithm 2 (i.e., ML or NNLS) has been analyzed in the previous work [6], [22]. Thus, we focus on the first stage of the algorithm. By (20), we know that selecting $K$ largest components in $\{y_k\}_{k=1}^{N}$ is equivalent to the one-step IHT in (12). In fact, For the one-step IHT given as $\gamma_1 \propto H_K(A^* u)$. Following Proposition 1 and Remark 1, we have

$$
E[A_k^* u] = \begin{cases}
\sum_{\ell \in S^2} \frac{D^2}{D + 1} \gamma^k_\ell, & \forall k \notin S^2, \\
\sum_{\ell \in S^2} \frac{D^2}{D + 1} \gamma^k_\ell + \frac{D^3}{D + 1} \gamma^k_k, & \forall k \in S^2,
\end{cases}
$$

where the expectation is taking over $A$, $H$ and $Z$. Then, for matrices $A$, $H$ and $Z$ that satisfy the Gaussianity assumptions above, we always have

$$
E[A_k^* u] \leq E[A_k^* u] - \frac{D^3}{D + 1} \gamma_{\min}, \quad \text{for } k_1 \notin S^2, k_2 \in S^2,
$$

Therefore, as long as $A_k^* u \to E[A_k^* u]$, there would be a successfully support recovery by one-step IHT because $\gamma_1 \propto H_K(A^* u)$. To ensure $A_k^* u \to E[A_k^* u]$, by the concentration inequalities, we need sufficiently large $D$ and $M$. Consider that $D$, $M$ are reasonably large and $K = K$. The following theorem shows that the estimation of one-step IHT satisfies $\sup \gamma_1 = S^2 \times$ overwhelming probability.

**Theorem 3:** Assume that $\{a_k\}_{k=1}^{N}$ are i.i.d. random variables drawn uniformly from the sphere of bounded radius $\sqrt{D} < \infty$. Let $\{y_k\}_{k=1}^{N}$ be defined in (19), $\hat{S}$ be the indices of $K$ largest components in $\{y_k\}_{k=1}^{N}$. Then we have

$$
P\left( \hat{S} = S^2 \right) \geq 1 - \epsilon - \frac{c_0}{D^{1/2}},
$$

provided that

$$
M \geq c_4 \left( \frac{c_5 K}{D} \frac{\sigma^2}{D \gamma_{\min}^2} \right)^2 \log \left( \frac{(eD)^2}{\epsilon} \right),
$$

$$
K \leq \frac{c_5 D}{c_4 \log(ND)} ,
$$

(24)

where $c_4, c_5$ are universal positive constants and $c_\gamma = \frac{\gamma_{\max}}{\gamma_{\min}}$. \qed

From Theorem 3, we see that there is $S^2 \subset \sup \gamma_1$ for $K \geq K$. Nevertheless, when taking the largest $K \geq K$ components in $\{y_k\}_{k=1}^{N}$, it is possible to detect $K = O(D^2)$ active users. Specially, when $K \to N$, the method is reduced to the standard ML estimator, which is able to detect $K = \tilde{O}(D^2)$ active users.

**Theorem 4** below shows that the estimated set $S^0$ in Algorithm 2 will contain all the indices in $S^2$ for the choice
\(\rho = 1\). Thus the first stage of Algorithm 2 is able to reduce the dimension of the coordinate-wise optimization.

**Theorem 4:** Under the same hypothesis in Theorem 3. Let \(\{\gamma_k\}_{k=1}^{N}\) be defined in (19), \(S^0\) be the set given by Algorithm 2 with \(\rho = 1\). Then we have

\[
P\left(S^0 \subset S^0\right) \geq 1 - \varepsilon - \frac{c_0}{D/V^2},
\]

provided that \(M\) satisfies inequality in (24) and \(K \leq \min\left\{\frac{c_0 D}{\varepsilon c_5 \log(DM)}, \frac{c_7 N}{c_5}\right\}\). Where \(c_6, c_7\) are universal positive constants.

**Proof:** See Appendix F. \(\square\)

Recall that Theorem 3 guarantees that \(S^0\) can be successfully recovered via one-step IHT when \(K = O(D)\). To ensure \(K = O(D^2)\) active users can be detected, we need a larger size of \(S^0\) by setting the parameter \(\rho\). When \(\rho\) is set to be around 1, numerical experiments suggest that the accelerated ML has similar performance compared to ML even for \(K > D\). In the next section, simulation results will be provided to validate the performance of the proposed Algorithm 2.

**V. SIMULATION AND DISCUSSIONS**

In this section, we present numerical results to compare the proposed Algorithm 2 with other existing algorithms. For convenience, we will refer Algorithm 2 as “AccML” (accelerated maximum-likelihood) in the remaining of this paper.

Column-normalized sub-sampled Fourier matrices (for a fair comparison with AMP [5], [18]) are used for the codebook \(A\) in (2), i.e., \(|a_k|^2 = 2\). The per-user SNR (for active users) is defined as

\[
\text{SNR} = \frac{\|a_{\pi,\alpha}\|^2 g_0^2 E\left[\|	ilde{h}_{1,\alpha}\|^2\right]}{E[\|Z\|^2_F]} = \frac{Dg_0^2 M}{DM \sigma^2} = \frac{g_0^2}{\sigma^2},
\]

where \(E[\|	ilde{h}_{1,\alpha}\|^2] = M\) since \(\tilde{h}_{1,\alpha} \in \mathbb{C}^M\) follows \(\mathcal{CN}(0, I_M)\). Then, the per-user \(E_b/N_0\) (energy per bit to noise power spectral density ratio) in defined as

\[
\frac{E_b}{N_0} = \frac{\text{SNR}}{R} = \frac{L Dg_0^2}{B \sigma^2},
\]

where \(R = \frac{B}{D_{\text{total}}} = \frac{B}{L D}\).

Per-user probability of error (PUPE) is used as the performance metric. By following Section II-B, the set of decoded messages is denoted as \(\bar{T}(Y) \equiv T(\mathcal{L}(Y))\). The set of messages \(\mathcal{L}(Y)\) is defined as \(\{x_1, x_2, x_3, \ldots\}\) and \(\mathcal{L}(Y)\) is a subset of \(Y\). For reference, the set of true messages of \(K\) users is denoted by \(W_a\) and \(|W_a| = K\). A miss detection error is declared if \(x_{\hat{r}} \notin \mathcal{L}(Y)\) and a false alarm error is declared if \(\hat{r} \in \mathcal{L}(Y) \setminus W_a\). Then, PUPE can be calculated as

\[
P_c = P_{md} + P_{fa},
\]

where \(P_{md} = 1 - \frac{E[|\mathcal{L}(Y) \cap W_a|]}{K}\), \(P_{fa} = \frac{E[|\mathcal{L}(Y) \setminus W_a|]}{K}\) are per-user probability of miss detection and false alarm, respectively.

**A. SPARC Decoding**

To evaluate the effectiveness of AccML for SPARC decoding, we consider a single section case \((L = 1, B = L \cdot J = J)\) with no tree code involved. Fig. 2a demonstrates the PUPE performance of AccML and Fig. 2b demonstrates the related computational complexity in terms of processing time in second. One-Step IHT in Algorithm 1, ML and NNLS [6] as well as AMP [18] and OAMP [21] are included for comparison. Moreover, another covariance-based approach call Bandit Sampling ML (BSML) [35, Algorithm 2] is added for comparison, where the initialization parameter is given by \(B_{BSML} = N/2 = 2^J/2\) and \(\epsilon_{BSML} = 0.6\) [35, Section VI]3. For simplicity, all LSFCs \(g_0^2\) are assumed to be 1 and \(K\) is known at the receiver. For AccML, \(\rho = 1\) and \(\rho = 1.05\) are included to show the effect of \(\rho\). For all covariance-base algorithm, including NNLS, ML, AccML in Algorithm 2 and BSML, the termination criteria is set to \(t_{\text{max}} = 10\) and \(\epsilon = 10^{-6}\).

As shown in Fig. 2a, both AMP and OAMP give poor PUPE performance for \(K > D\). We must emphasize that, according to [6], and [22], up to \(K = O(D)\) active users can be successfully recovered by AMP and OAMP under the compressive sensing framework. Since the current setup has \(K \gg D\) active users, it is beyond the recovery capability of AMP-type algorithms. Actually, both AMP and OAMP suffer from instability issue and results in early termination [5].

When ML is combined with the one-step thresholding, the resulting AccML can achieve the similar PUPE performance as ML and BSML in Fig. 2a but has much lower computational complexity as shown in Fig. 2b. The computational complexity of ML is \(O(N.D^2)\), which increases linearly with \(N = 2^J\) and thus exponentially with \(J\). AccML can reduce the dimension of the problem with the one-step thresholding-based algorithm and the reduction effect increases with \(J\) as shown in Fig. 2b.

Comparing \(\rho = 1\) and \(\rho = 1.05\) for AccML, a larger \(\rho\) means that fewer support indices are selected by the thresholding-based algorithm, which induces larger PUPE as shown in Fig. 2a. Meanwhile, a larger \(\rho\) means that more dimensions are reduced for ML and thus results in more computational complexity reduction as shown in Fig. 2b. More specifically, as shown in Fig. 2c, the AccML reduced the size of the support indices to \(K_{\text{acc}}^\rho < \frac{1}{2}\) and \(K_{\text{acc}}^\rho < \frac{1}{4}\) for \(\rho = 1\) and \(\rho = 1.05\), respectively. This is consistent with the results in Fig. 2b, where the processing time is reduced accordingly.

**B. Unsourced Random Access**

The performance of AccML is further evaluated for URA. For a fair comparison, the setup in [6, Fig. 8(a)] is selected. However, the thresholding parameter for the decision of support is not provided in details in [6]. Equivalently, we select the indices of largest \(T\) components of \(\gamma\) as the support. For simplicity and fairness, we select \(T = K\) and \(T = K + 100\) in this example. Finally, for AccML initialization, we set \(\rho = 1.05\).

---

3 The initialization parameter \(\alpha\) and \(\beta\) for the Thompson Sampling ML [35, Algorithm 3] is not given clearly. Considering that its effect on computational complexity reduction is fundamentally similar to the BSML, we omit its simulation results here.
Fig. 2. (a) PUPE as a function of $M$ where $K = 300$, $D = 120$, $J = 12$, $N = 2^J = 4096$ at SNR $= -10$ dB, $E_b/N_0 = 0$ dB; (b) Processing Time(s) as a function of $J = \log_2 N$ where $K = 300$, $M = 300$, $D = 120$ at SNR $= -10$ dB, $E_b/N_0 = 0$ dB. AMP, OAMP and One-Step IHT are omitted due to their poor performance in Fig. 2a; (c) $K_{sel}/N$ as a function of $J$.

Figs. 3a and 3b compare the PUPE performance of ML and AccML. AccML can achieve similar PUPE performance as ML for $E_b/N_0 \leq 2$ dB in Fig. 3a and $E_b/N_0 \leq -2$ dB in Fig. 3b but the gap becomes larger as $E_b/N_0$ increases. However, for URA in mMTC, the PUPE of practical interest is around 0.05 and the active users are generally working at low $E_b/N_0$ due to power limitation. In short, AccML performs remarkably well as ML for PUPE around 0.05.

Consider that the BS has $M = 300$ antennas. This setup could support URA for $K = 300$ active users with ML or AccML algorithm. However, the number of antennas available at BS is around 64 to 128 for most current massive MIMO systems. Therefore, the maximum active user can be supported is also around 100. In the next subsection, we consider a more practical setup for URA.

C. Practical MIMO-OFDM Scenarios

We consider physical uplink shared channel (PUSCH) in a MIMO orthogonal frequency division multiplexing (OFDM) system. A total of 12 resource blocks (RB) are allocated for transmission and each RB contains 12 sub-carriers with sub-carrier spacing of 15 kHz [36, Clause 4]. Each sub-carrier contains 14 channel uses (or OFDM symbols), where the last 10 OFDM symbols are used for PUSCH and the remaining 4 OFDM symbols are reserved for other usages such as control signal [37, Clause 11]. Each section occupies a single RB of 120 channel uses. Channel is assumed to stay static within each RB and may change randomly across different RB.

In summary, a total of $D_{total} = 1440$ channel uses are used for URA, which are divided into $L = 12$ sections (corresponds to 12 RB) with $D = 120$ channel uses each. Consider that $K = [50, 75, 100, 125, 150]$ active user are supported by BS with $M = 64$ antennas. Each user transmit $B = 50$ information bits across $L = 12$ sections. Each section is augmented to size $J = 12$ with the additional parity bits, which are allocated as $0, 7, 8, \ldots, 8, 11, 12$. For the SPARC decoding, we set $T$ as $K + K_\Delta$ with $K_\Delta = 50$ and $\rho = 1.05$ for AccML initialization. For simplicity, the details
of standard OFDM operations involving adding/stripping off cyclic prefix and the IFFT/FFT operation are omitted.

The solid and dot lines in Fig. 4a show the PUPE performance versus $E_b/N_0$ for ML and AccML, respectively. Overall, AccML achieves the similar performance as ML. When the number of the active users increases, the performance of AccML degrades slightly. Intuitively, the hard-thresholding of step 1 in Algorithm 2 may miss some non-zero entries in $\gamma$, which results in lager PUPE.

Fig. 4b shows the required $E_b/N_0$ to achieve $\text{PUPE} \leq 0.05$ versus different number of active users. For comparison, AMP, OAMP, BSML and NNLS are also included. Finally, the asymptotic performance analysis based on the receiver operating characteristic (ROC) of ML [38], [39] is added for reference, which is denoted as “ROC”. As shown in Fig. 4b, AccML achieves the similar performance as ML and BSML and significantly outperforms NNLS, AMP and OAMP. In particular, for $K \geq 100$, NNLS, AMP and OAMP deteriorate dramatically but AccML still performs near optimally. It should be highlighted that both AMP and OAMP work under extremely poor condition for $K \geq 100$, which results in early termination. Overall, in terms of both $E_b/N_0$ requirement and the scaling law in $K$ and $D$, the AccML outperforms AMP and OAMP significantly.

The processing time of both SPARC and tree decoding is shown in Fig. 4c. For SPARC decoding, the computational complexity is summarized in Table I. The computational complexity of ML and NNLS in general is $O(D^2 \cdot N)$ [6] but NNLS use less calculation in each iteration. The computational complexity of BSML is $O(D^2 \cdot \frac{N+BB_{BSML}}{2})$ where $BB_{BSML} = \frac{N^2}{2}$ according to [35, Section VI]. The computational complexity of AccML is reduced to $O(D^2 \cdot K_{sel})$ where $K_{sel} \leq \frac{N}{4}$ according to Fig. 2c. Finally, the computational complexity of AMP and OAMP in general is $O(M \cdot N \log N)$ but AMP use less calculation in each iteration. We can see that AccML could reduce the computational complexity of the ML stage significantly. As shown shown in Fig. 4c, the processing time of AccML is much smaller than ML, BSML and NNLS. In particular, when $K > 100$, AccML only consumes slightly larger time than AMP but achieve the near optimal performance as ML and BSML. It demonstrates the advantages of the proposed two-stage AccML over all existing methods.

VI. CONCLUSION

In this paper, we propose a two-stage method for SPARC decoding that combines a one-step thresholding-based algorithm for dimension reduction and ML for refinement. Due to the dimension reduced in the first stage, the complexity of ML is significantly reduced while the overall performance still being maintained. Adequate simulation results are provided to validate the near-optimal performance and the low computational complexity. In addition, theoretical analyses are given for both the covariance-based sparse recovery model and the thresholding algorithm. When convex relaxation is
considered, the upper bound for the number of active users is
given and proven to be optimal. For the one-step thresholding-
based algorithm, theoretical analyses show that it is highly
possible to contain all the active users when $M$ and $D$
are reasonably large. For future works, the extension of the
proposed algorithm and theoretical analyses to more practical
spatially correlated channel models seems promising.

**Appendix A**

**Key Lemmas**

**Lemma 1:** Assume that $a_k \in \mathbb{C}^D$, $k \in [N]$ are i.i.d. random
variables drawn uniformly from the sphere of radius $\sqrt{D}$. Then
for all $k \in [N]$, $i, j \in [D]$ there is

$$E[|a_{k,i}|^2 | a_{k,j}|^2] = \frac{D}{D + 1}, \quad \forall i \neq j,$$

and

$$E[|a_{k,i}|^4] = \frac{2D}{D + 1}, \quad \forall i \in [D].$$

**Proof:** For all $k \in [N]$, define the real-valued vector

$$a_k^R := [\text{Re}(a_k), \text{Im}(a_k)] \in \mathbb{R}^{2D},$$

where $\text{Re}(a_k)$ and $\text{Im}(a_k)$ are the real part and imaginary
part of $a_k$, respectively. Then $(a_k^R)_{k=1}^N$ are also i.i.d. random
variables drawn uniformly from the sphere of radius $\sqrt{D}$. And we have

$$E[|a_{k,i}|^2 | a_{k,j}|^2] = E[|(\text{Re}(a_{k,i})),(\text{Im}(a_{k,j}))|^2|(|\text{Re}(a_{k,j})),(|\text{Im}(a_{k,j}))|^2]$$

$$= 4E[|a_{k,i}|^2 | a_{k,j}|^2].$$

(28)

As $\frac{a_k}{\sqrt{D}}$ are i.i.d. random variables drawn uniformly from the
unit sphere, we rewrite the vector $\frac{a_k}{\sqrt{D}} := \frac{g_k}{\|g_k\|_2}$, where $g_k \in \mathbb{R}^{2D}$ with each component of $g_k$ be i.i.d. standard Gaussian. Then

$$\frac{1}{D}E[|a_{k,i}|^2 | a_{k,j}|^2] = E\left[\left|\frac{g_{k,i}}{\|g_k\|_2} \cdot \frac{g_{k,j}}{\|g_k\|_2}\right|^2\right]$$

$$= E\left[\left|g_{k,i} \cdot g_{k,j}\right|^2 \int_0^{t=\infty} e^{-t\|g_k\|_2^2}dt\right]$$

$$= E\left[\int_0^{t=\infty} g_{k,i}^2 e^{-t\|g_k\|_2^2} g_{k,j}^2 e^{-t\|g_k\|_2^2} \prod_{s \neq i,j} e^{-t\|g_k\|_2^2}dt\right]$$

$$= \int_0^{t=\infty} [\tilde{f}(t)]^2 [\tilde{f}(t)]^{2(D-2)}dt,$$

where $\tilde{f}(t) = E\left[e^{-t\|g_k\|_2^2}\right] = \frac{1}{\sqrt{2\pi t}}$, and $\tilde{f}(t) = -(1 + 2t)^{3/2}$. Thus,

$$E[|a_{k,i}|^2 | a_{k,j}|^2] = 4D \int_0^{t=\infty} [\tilde{f}(t)]^2 [\tilde{f}(t)]^{2(D-2)}dt$$

$$D \int_0^{t=\infty} (1 + 2t)^{-D-2}dt = \frac{D}{4D + 4}.$$}

Therefore, by (28) we have

$$E[|a_{k,i}|^2 | a_{k,j}|^2] = 4E[|a_{k,i}|^2 | a_{k,j}|^2] = \frac{D}{D + 1}.$$ 

Now we compute $E[|a_{k,i}|]$. By $|a_{k,i}|^2 = D^2$, we have

$$D^2 = \sum_{i \neq j} E[|a_{k,i}|^2 | a_{k,j}|^2] + \sum_{i=1}^D E[|a_{k,i}|^4]$$

$$= D(D-1)E[|a_{k,i}|^2 | a_{k,j}|^2] + DE[|a_{k,i}|^4],$$

for all $i \in [D]$, and $i \neq j$. Thus we have

$$E[|a_{k,i}|] = D - (D-1)E[|a_{k,i}|^2 | a_{k,j}|^2] = \frac{2D}{D + 1}.$$ 

This completes the proof.

**Lemma 2 ([40, Proposition 5]):** Let the vectors $x, y \in \mathbb{R}^n$
be sampled independently and uniformly from the unit sphere $\mathbb{S}^{n-1}$, and $\theta$ be the angle between $x$ and $y$. Then,

$$P\left(\theta - \frac{\pi}{2} \geq \varepsilon\right) \leq \tilde{c}_0 \sqrt{n}(\cos \varepsilon)^{n-2},$$

for all $n \geq 2$ and $\varepsilon \in (0, \frac{\pi}{2})$, where $\tilde{c}_0$ is an universal constant.

**Lemma 3:** Assume that $\{a_k^R\}_{k=1}^N$ are i.i.d. random variables
drawn uniformly from the sphere of radius $\sqrt{D}$. For any non-negative $\gamma^2$ supported on $\mathbb{S}^2$, if provided $\infty > D > 8 \log(DN)$, then it holds with probability at least $1 - \frac{c_0}{\sqrt{D}}$ that

$$\frac{1}{D}a_k^R a_{\ell} \leq \sqrt{32 \log(DN)/D}, \quad \forall k \neq \ell,$$ 

and

$$\sum_{i=1}^D \sum_{i=1}^D |a_{k,i} a_{\ell,i}|^2 \leq \sum_{i=1}^D 32\gamma^2 D \log(DN),$$

(30)

for any $k \in \mathbb{S}^2$, where $\gamma^2$ is an universal constant.

**Proof:** Recall that $a_k^R = [\text{Re}(a_k), \text{Im}(a_k)]$. We also
denote

$$a_k^R := [\text{Im}(a_k), -\text{Re}(a_k)] \in \mathbb{R}^{2D} \quad k \in [N].$$

Then by definition we have

$$\frac{1}{D}a_k^R a_{\ell} = \frac{1}{D} [\text{Re}(a_k) \cdot \text{Re}(a_{\ell}) + \text{Im}(a_k) \cdot \text{Im}(a_{\ell})]$$

$$- i \cdot (\text{Re}(a_k) \cdot \text{Re}(a_{\ell}) - \text{Re}(a_k) \cdot \text{Im}(a_{\ell}))$$

$$= \frac{1}{D} \left| (a_k^R)^T a_{\ell} - (a_k^R)^T a_{\ell}^R \right|.$$ 

(32)

Notice that for $k \neq \ell$, $a_k^R \in \mathbb{R}^{2D}$ and $a_{\ell}^R \in \mathbb{R}^{2D}$ are sampled independently and uniformly from the unit sphere $\mathbb{S}^{2D-1}$. Let $\theta_{k,\ell}$ be the angle between $a_k^R$ and $a_{\ell}^R$. Then, for some given $k$ and $\ell$ such that $k \neq \ell$, by Lemma 2 we have

$$P\left(\theta_{k,\ell} - \frac{\pi}{2} \geq \varepsilon\right) \leq \tilde{c}_1 \sqrt{2D} (\cos \varepsilon)^{2D-2}$$

for any $\varepsilon \in (0, \frac{\pi}{2})$ and some universal constant $\tilde{c}_1$. Thus, for any $k \neq \ell$, by a union bound we have

$$P\left(\theta_{k,\ell} - \frac{\pi}{2} \geq \varepsilon\right) \leq \tilde{c}_1 (\frac{N}{2}) \sqrt{2D} (\cos \varepsilon)^{2D-2}.$$
For any \( \varepsilon \in (0, 1) \), by \( \cos \varepsilon \leq 1 - \frac{\varepsilon^2}{2} + \frac{\varepsilon^4}{24} \leq 1 - \frac{\varepsilon^2}{2} \), we have for some universal constant \( c_0 \),
\[
P\left( \left| \theta_{k,\ell} - \frac{\pi}{2} \right| \geq \sqrt{8\log(DN)/D} \right) \leq c_0/(2D\sqrt{D}), \quad \forall k \neq \ell.
\]
Therefore, with probability at least \( 1 - \frac{c_0}{2D\sqrt{D}} \) we have \( (D < \infty) \)
\[
|\cos \theta_{k,\ell}| \leq \left| \cos \left( \frac{\pi}{2} + \sqrt{8\log(DN)/D} \right) \right| \leq \sqrt{8\log(DN)/D}, \quad \forall k \neq \ell.
\]
By the definition of \( \theta_{k,\ell} \), we then have with probability at least \( 1 - \frac{c_0}{2D\sqrt{D}} \) it holds
\[
\left( a_k^R \right)^T a_{\ell}^R / D = |\cos \theta_{k,\ell}| \leq \sqrt{8\log(DN)/D}, \quad \forall k \neq \ell.
\]
Similarly, with probability at least \( 1 - \frac{c_0}{2D\sqrt{D}} \) we also have
\[
0 \leq \left( a_k^R \right)^T a_{\ell}^R / D \leq \sqrt{8\log(DN)/D}, \quad \forall k \neq \ell.
\]
Then, together with (32), it yields
\[
|a_k^\ast a_\ell| \leq \left( a_k^R \right)^T a_{\ell}^R + \left( a_k^R \right)^T a_{\ell}^R \leq 4\sqrt{2D\log(DN)}
\]
with probability at least \( 1 - \frac{c_0}{2D\sqrt{D}} \). Therefore, for \( k \notin S^2 \) and \( \ell \in S^2 \) we must have \( k \neq \ell \) and thus
\[
\sum_{i=1}^D \gamma_i^2 \left| \bar{a}_{k,i} a_{\ell,i} \right|^2 = \sum_{i=1}^D \gamma_i^2 D^2 \left| a_k^\ast a_\ell \right|^2 \leq \sum_{i=2}^D 32\gamma_i^2 D \log(DN).
\]
This completes the proof.

**Lemma 4:** Let \( \mu \) be the mutual coherence of \( \{a_k\}_{k=1}^N \) defined in (21), and \( A \) be defined in (9) whose columns are denoted by \( \{A_k\}_{k=1}^N \). Then, the coherence of \( A \), i.e., \( \tilde{\mu} = \max_{k \neq \ell} \frac{\left| a_k^\ast a_\ell \right|}{\mu} \), satisfies \( \tilde{\mu} = \mu \).

**Proof:** By the definition of \( A \) and \( A_k \), for all \( k, \ell \in [N] \) we have
\[
|\langle A_k, A_\ell \rangle| = |\langle \text{vec}(a_k^\ast), \text{vec}(a_\ell^\ast) \rangle| = \left( \sum_{1 \leq i,j \leq D} a_{k,i} a_{\ell,j} \right)^2 = |\langle a_k^\ast a_\ell \rangle|^2.
\]
By taking \( k = \ell \) in the above equation, we also have \( \|A_k\|_2 = \|a_k\|_2 \), \( \forall k \in [N] \). Thus, by the definition of \( \mu \) and \( \tilde{\mu} \), we have \( \mu = \tilde{\mu} \).

**Lemma 5** (8, Theorem 11]): Let \( \Delta = \hat{\Sigma}_y - \Sigma_y \) where \( \hat{\Sigma}_y \) and \( \Sigma_y \) are given by (4) and (3) respectively. Let \( \varepsilon > 0 \) and \( \|a_k\|_2^2 = D, \; k \in [N] \), then for some universal constant \( c \), it holds that
\[
\|\Delta\|_F \leq \frac{D \left( \left\| \gamma^2 \right\|_1 + \sigma^2 \right)^{1/2}}{\sqrt{M}} \sqrt{c \log \left( \frac{(eD)^2}{\varepsilon} \right)}
\]
with probability at least \( 1 - \varepsilon \) provided \( M \geq c \log \left( \frac{2(eD)^2}{\varepsilon} \right) \).

**APPENDIX B**

**Proof of Proposition 1**

**Proof:** By the definition of \( \mathcal{Y}_k \) and \( \hat{\Sigma}_y \), we have
\[
\mathcal{Y}_k = \frac{1}{M} \sum_{i,j \in [D]} \tilde{a}_{k,i} a_{j,i} y(i,:) Y(j,:)^\ast
\]
\[
= \frac{1}{M} \sum_{i,j \in [D]} \pi_{k,i} a_{j,i} \left( \sum_{\ell \in [N]} \sqrt{\gamma^2 \ell a_{\ell,i} h^\ast_{\ell} + z^i_{\ell}} \right)
\]
\[
\left( \sum_{m \in [N]} \sqrt{\gamma^2 m \pi_{m,j} h_m + z^j_{\ell}} \right).
\]
The assumption is that \( \{a_k\}_{k=1}^N \) are independent from channel vectors and AWGN. Then, by the independence of \( \{h_k\}_{k=1}^N \), \( \{z_j\}_{j=1}^D \) and \( \{a_k\}_{k=1}^N \), we have
\[
E[\mathcal{Y}_k]
\]
\[
= \frac{1}{M} \sum_{i,j \in [D]} \sum_{\ell, m \in [N]} E[\pi_{k,i} a_{j,i}] y(i,:) Y(j,:)^\ast
\]
\[
+ \frac{1}{M} \sum_{i,j \in [D]} E[\pi_{k,i} a_{j,i}] E[z^i_{\ell} z^j_{\ell}]
\]
\[
= \sum_{i,j \in [D]} E[\pi_{k,i} a_{j,i}] y(i,:) Y(j,:)^\ast
\]
\[
\sum_{i \neq j} \sum_{\ell, m \in [N]} E[\pi_{k,i} a_{j,i}] y(i,:) Y(j,:)^\ast
\]
\[
\sum_{i \neq j} \sum_{\ell, m \in [N]} E[\pi_{k,i} a_{j,i}] y(i,:) Y(j,:)^\ast
\]
\[
\sum_{i=1}^D \sum_{\ell \in S_i^\ast} E[\pi_{k,i} a_{j,i}] y(i,:) Y(j,:)^\ast
\]
\[
+ \sum_{i=1}^D E[|a_{k,i}|^2] \sigma^2,
\]
where the last equation follows from the fact that \( \gamma^2 \) is supported on \( S^2 \). Assume that \( \{a_k\}_{k=1}^N \) are i.i.d. random variables drawn uniformly from the sphere of radius \( \sqrt{D} \). Then for \( k \in S^2 \), there is
\[
E[\mathcal{Y}_k]
\]
\[
= \sum_{i=1}^D \sum_{\ell \in S_i^\ast} E[|a_{k,i}|^4] \gamma^2 \ell
\]
\[
+ \sum_{i=1}^D \sum_{\ell \in S_i^\ast} E[|a_{k,i}|^2] |a_{\ell,i}|^2 \gamma^2 \ell
\]
\[
+ \sum_{i=1}^D \sum_{\ell \in S_i^\ast} E[|a_{k,i}|^2] \sigma^2
\]
\[
= \sum_{i=1}^D \sum_{\ell \in S_i^\ast} \frac{2D}{D+1} \gamma^2 \ell
\]
\[
+ \sum_{i=1}^D \sum_{\ell \in S_i^\ast} \frac{D}{D+1} \gamma^2 \ell + \sigma^2
\]
\[
= \sum_{i=1}^D \frac{D^2}{D+1} \gamma^2 \ell + \frac{D^3}{D+1} \gamma^2 \ell + \sigma^2,
\]
the second equation follows from Lemma 1. Also, for \( k \notin S^3 \), we know \( k \neq \ell \), \( \forall \ell \in S^3 \), and thus by Lemma 1 we have
\[
E[\gamma_k] = \sum_{i=1}^{D} \sum_{\ell \in S^3} E[|a_{k,i}|^2 | a_{\ell,i}|^2] \gamma_{\ell}^2 + \sum_{i \neq j, \ell \in S^3} E[|a_{k,i} a_{k,j} a_{\ell,i} a_{\ell,j}|] \gamma_{\ell}^2 + D \sigma^2 = \sum_{\ell \in S^3} \frac{D^2}{D+1} \gamma_{\ell}^2 + D \sigma^2.
\]
Thus we complete the proof. \( \square \)

APPENDIX C

PROOF OF THEOREM 1

The proof of the theorem relies on the so-called robust null space property (NSP) of \( \mathcal{A} \) and a substantial lemma in [30]. The robust NSP is described in the following definition.

Definition 1 (Robust NSP, [30, Definition 4.17]): The matrix \( \mathcal{A} \in \mathbb{C}^{D \times N} \) is said to satisfy the robust \( \ell_2 \)-NSP of order \( s \) with constants \( 0 < \beta < 1 \) and \( \tau > 0 \) if
\[
\| \mathcal{A} v \|_1 \leq \beta \| v \|_2 + \tau \| \mathcal{A} v \|_2, \quad \forall v \in \mathbb{R}^N
\]
holds for any subset \( S \subseteq [N] \), \( |S| \leq s \). \( \mathcal{S} \) denotes the complement of \( S \) in \( [N] \).

Now we give the proof of theorem 1.

Proof: Using the notation in (10), the problem (8) can be equivalently written as
\[
\min_{\gamma \in \mathbb{R}^N} \| \gamma \|_1, \quad \text{s.t.} \quad \| \mathcal{A} \gamma \|_2 \leq \eta. \tag{34}
\]
Now define a centered version of \( \mathcal{A} \) denoted by \( \overline{\mathcal{A}} \), where \( \overline{\mathcal{A}}_k \in \mathbb{C}^{D(D-1)} \) (i.e., the \( k \)-th column of \( \overline{\mathcal{A}} \)) is formed by the vectorization of non-diagonal elements of \( a_k a_k^* \), and thus \( \overline{\mathcal{A}} \in \mathbb{C}^{D(D-1) \times N} \). According to [6, Theorem 2] and [6, Theorem 10], we know there exists universal constants \( \bar{c}_2, \bar{c}_3 \), such that as long as
\[
2K \leq \frac{\bar{c}_2 D(D-1)}{\log^2 (e N / D)}
\]
then with probability at least \( 1 - e^{-\bar{c}_3 D} \) it holds that
\[
\| v \|_1 \leq 0.68 \| v \|_2 + \frac{3}{D} \| \overline{\mathcal{A}} v \|_2, \quad \forall v \in \mathbb{R}^N
\]
for any subset \( S \subseteq [N] \), \( |S| \leq K \). It means \( \overline{\mathcal{A}} \) satisfies the robust \( \ell_2 \)-NSP of order \( K \) under the condition above. Further, by the definition of \( \overline{\mathcal{A}} \), we know \( \| \overline{\mathcal{A}} v \|_2 \geq \| \mathcal{A} v \|_2 \), thus \( \mathcal{A} \) also satisfies the robust \( \ell_2 \)-NSP of order \( K \) with constants \( 0.68 \) and \( \frac{3}{D} \). Therefore, by [30, Theorem 4.19] we have
\[
\| \gamma - \gamma_i \|_1 \leq \frac{12D}{1 - 0.68} \eta \leq \frac{12}{\sqrt{M}} \eta \leq \frac{37.5 (\| \gamma \|_1 + \sigma^2)}{\sqrt{M}} \sqrt{c \log \left( \frac{(eD)^2}{\varepsilon} \right)},
\]
where the first inequality follows from \( \sigma (\gamma^*)_1 = 0 \) as \( \| \gamma \|_1 \leq K \), the last inequality follows from Lemma 5. Recall that \( c \) is a universal constant. By setting \( c_1 = 37.5 \sqrt{c} \), then the proof is completed. \( \square \)

APPENDIX D

PROOF OF THEOREM 2

Before the proof, we recall the definition of restricted isometry constant and a lemma for the convergence guarantees of IHT algorithm in [30].

Definition 2: The s-th restricted isometry constant (denoted \( \delta_s \)) of the matrix \( \mathcal{B} \in \mathbb{C}^{D^2 \times N} \) is the smallest \( \delta > 0 \) such that
\[
(1 - \delta) \| v \|_2^2 \leq \| \mathcal{B} v \|_2^2 \leq (1 + \delta) \| v \|_2^2
\]
for all s-sparse vectors \( v \in \mathbb{C}^N \).

Lemma 6 ([30, Theorem 6.18]): Suppose that the \( 3s \)-th restricted isometry constant of the matrix \( \frac{\mathcal{A}}{\sqrt{N}} \in \mathbb{C}^{D \times N} \) satisfies \( \delta_{3s} \leq \frac{\delta}{\sqrt{N}} \). Then, for any s-sparse \( \gamma^2 \) and \( \delta \) in (10) with \( \| \delta \|_2 \leq \eta \), the sequence \( \{ \gamma_t \}_{t \geq 0} \) defined by IHT in (12) with some constant step size \( \alpha \) satisfies
\[
\| \gamma_{t+1} - \gamma^2 \|_2 \leq \sqrt{3} \delta_{3s} \| \gamma_t - \gamma^2 \|_2 + \frac{2.18}{1 - \sqrt{3} \delta_{3s}} \eta, \forall t \geq 0.
\]
Now we give the proof of theorem 2.

Proof: Let \( \mu \) be the mutual incoherence of \( \mathcal{A} \). By the proof in Lemma 4, we know \( \| \mathcal{A} \| \leq D \), and thus \( \mathcal{A} \) has \( \ell_2 \)-normalized columns. By [30, Theorem 5.3] and [30, Eq. (5.2)] we have
\[
(1 - (3K - 1)\mu) \| v \|_2^2 \leq \| \mathcal{A} v / D \|_2^2 \leq (1 + (3K - 1)\mu) \| v \|_2^2
\]
for all vectors in \( \{ v \in \mathbb{C}^N : \| v \|_0 \leq 3K \} \). As long as the mutual incoherence of the matrix \( \mathcal{A} = [a_1, a_2, \ldots, a_N] \) satisfies \( \mu \leq \frac{1}{3 \sqrt{3(K-1)}} \), then by Lemma 4, we know the mutual incoherence of \( \frac{\mathcal{A}}{\sqrt{N}} \) satisfies \( \mu \leq \frac{1}{3 \sqrt{3(K-1)}} \). Therefore, by (35) we have
\[
(1 - \frac{1}{\sqrt{3}}) \| v \|_2^2 \leq \frac{1}{D} \| \mathcal{A} v \|_2^2 \leq (1 + \frac{1}{\sqrt{3}}) \| v \|_2^2
\]
for all vectors in \( \{ v \in \mathbb{C}^N : \| v \|_0 \leq 3K \} \). By the definition of the restricted isometry constant in Definition 2, we have \( \delta_{3K} \leq \frac{1}{\sqrt{3}} \). Recall that \( \gamma^2 \) is \( K \)-sparse. By Lemma 6, we then complete the proof. \( \square \)

APPENDIX E

PROOF OF THEOREM 3

Proof: By simple reformulation, we have
\[
\gamma_k = \sum_{1 \leq i,j \leq D} \tilde{a}_{k,i} a_{k,j} \left( \sum_{\ell \in S^3} a_{\ell,i} a_{\ell,j} \gamma_{\ell}^2 \right) + \sum_{1 \leq i,j \leq D} \tilde{a}_{k,i} a_{k,j} \Delta(i,j)
\]
\[
= \sum_{\ell \in S^3} \frac{D}{c} \tilde{a}_{k,\ell} a_{\ell,i} \left( \sum_{\ell \in S^3} a_{\ell,i} a_{\ell,j} \gamma_{\ell}^2 \right) + A_k^* \delta. \tag{36}
\]
Recall that in (7) we have defined \( \eta = \frac{(\sum_{\ell \in S^3} \gamma_{\ell}^2 + \sigma^2)}{\sqrt{M}} \sqrt{c \log \left( \frac{(eD)^2}{\varepsilon} \right)} \). Then, for the later term in (36), we have
\[
|A_k^* \delta| \leq \| A_k \|_2 \| \delta \|_2 \leq D \eta, \quad \forall k \in [N]
\]
with probability at least $1 - \varepsilon$ provided $M \geq c \log \left( \frac{(eD)^2}{\varepsilon} \right)$. The first inequality follows from Cauchy-Schwarz inequality. The last inequality follows from Lemma 5 and the fact that $\|A_k\|_2 = D$. Therefore, $\forall k_1 \in S^2$, we have,

$$\mathcal{Y}_{k_1} = \gamma_{k_1}^2 \left[ \sum_{i=1}^{D} |a_{i,j}|^2 \right] + \sum_{\ell \in S^1, \ell \neq k_1} \gamma_{\ell}^2 \left[ \sum_{i=1}^{D} \pi_{k_1,i} a_{\ell,i} \right]^2 + A_{k_1} \delta$$

$$\geq \gamma_{k_1}^2 D^2 - D\eta.$$  

(37)

Also, by (36) and Lemma 3 we have

$$\mathcal{Y}_{k_2} \leq \sum_{\ell \in S^1} 32 \gamma_{\ell}^3 D \log(DN) + D\eta, \ \forall k_2 \notin S^3$$

(38)

holds with probability at least $1 - \frac{c_0}{D^4}$. Then, we must have

$$\mathcal{Y}_{k_1} > \mathcal{Y}_{k_2} \text{ for any } k_1 \in S^2 \text{ and any } k_2 \notin S^3,$$

as long as

$$0 < \frac{1}{2} \gamma_{k_1}^3 - \sum_{\ell \in S^1} \gamma_{\ell}^3 16 \log(DN)/D - \frac{\eta}{D}, \ \forall k \in [N].$$  

(39)

To guarantee (39), we let

$$D \geq \frac{\tilde{c}_4}{\min_{\ell \in S^1} \gamma_{\ell}^3 \log(DN)}$$

and

$$M \geq c_4 \left( \sum_{\ell \in S^1} \gamma_{\ell}^3 + \gamma_2^3 / \min_{\ell \in S^1} \gamma_{\ell}^3 \right)^2 \log \left( \frac{(eD)^2}{\varepsilon} \right),$$

(40)

for some universal constant $\tilde{c}_4$ and $c_4$. Therefore, (39) is satisfied under the conditions in (40), and thus $S = S^3$. By noticing that

$$\frac{\gamma_{\ell}^2 \gamma_i^2 + \gamma_i^2}{D \gamma_{\ell}^2} \leq \frac{\max_{\ell} k \gamma_{\ell}^2 + 2 \gamma_i^2}{D \gamma_{\ell}^2},$$

we complete the proof. \hfill \Box

APPENDIX F

PROOF OF THEOREM 4

Proof: Similar to (37), by (36), Lemma 3 and triangle inequality we have

$$\mathcal{Y}_{k_1} \leq \gamma_{k_1}^2 D^2 + \sum_{\ell \in S^1} 32 \gamma_{\ell}^3 D \log(DN) + D\eta, \ \forall k_1 \in S^2.$$  

Then by the above inequality together with (38), we have

$$\frac{1}{N} \sum_{k=1}^{N} \mathcal{Y}_{k} \leq \frac{D^2}{N} \sum_{k=1}^{N} \gamma_{k}^2 + \sum_{\ell \in S^1} 32 \gamma_{\ell}^3 D \log(DN) + D\eta.$$  

Then, using the same argument to the proof of Theorem 3 in Appendix E, we have $\mathcal{Y}_{k} > \frac{1}{N} \sum_{k=1}^{N} \mathcal{Y}_{k}, \forall k \in S^1$, provided that $M$ satisfies (40) and $K \leq \min \left\{ \frac{c_6 D}{\log(DN)}, \frac{c_7 D}{\eta} \right\}$ for some sufficiently small universal constants $c_6, c_7$ and $c_8 = \frac{\max_{\ell \in S^1} k \gamma_{\ell}^2 + 2 \gamma_i^2}{D \gamma_{\ell}^2}$. Thus, we must have $S^2 \subset S^0$. \hfill \Box

REFERENCES

[1] E. D. Carvalho, E. Björnson, J. H. Sørensen, E. G. Larsson, and P. Popovski, “Random pilot and data access in massive MIMO for machine-type communications,” IEEE Trans. Wireless Commun., vol. 16, no. 12, pp. 7703–7717, Dec. 2017.

[2] A. C. Cirik, N. M. Balasubramanya, L. Lampe, G. Vos, and S. Bennett, “Toward the standardization of grant-free operation and the associated NOMA strategies in 3GPP,” IEEE Commun. Standards Mag., vol. 3, no. 4, pp. 60–66, Dec. 2019.

[3] W. Zhan and L. Dai, “Massive random access of machine-to-machine communications in LTE networks: Modeling and throughput optimization,” IEEE Trans. Wireless Commun., vol. 17, no. 4, pp. 2771–2785, Apr. 2018.

[4] Y. Poliansky, “A perspective on massive random-access,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2017, pp. 2523–2527.

[5] A. Fengler, P. Jung, and G. Caire, “SPARCs for unsourced random access,” IEEE Trans. Inf. Theory, vol. 67, no. 10, pp. 6894–6915, Oct. 2021.

[6] A. Fengler, S. Haghjighatashoor, P. Jung, and G. Caire, “Non-Bayesian activity detection, large-scale fading coefficient estimation, and unsourced random access with a massive MIMO receiver,” IEEE Trans. Inf. Theory, vol. 67, no. 5, pp. 2925–2951, May 2021.

[7] M. Mohammadkarimi, O. A. Dobre, and M. Z. Win, “Massive uncoordinated multiple access for beyond 5G,” IEEE Trans. Wireless Commun., vol. 21, no. 5, pp. 2969–2986, May 2022.

[8] C. Bockelmann et al., “Massive machine-type communications in 5G: Physical and MAC-layer solutions,” IEEE Commun. Mag., vol. 54, no. 9, pp. 59–65, Sep. 2016.

[9] D. Tse and P. Viswanath, Fundamentals of Wireless Communication. Cambridge, U.K.: Cambridge Univ. Press, 2005, Ch. 2.

[10] O. Ordentlich and Y. Poliansky, “Low complexity schemes for the random access Gaussian channel,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2017, pp. 2528–2532.

[11] V. K. Amalladinne, J. F. Chamberland, and K. R. Narayanan, “A coded compressed sensing scheme for unsourced multiple access,” IEEE Trans. Inf. Theory, vol. 66, no. 10, pp. 6509–6533, Jul. 2020.

[12] V. K. Amalladinne, A. K. Pradhan, C. Rush, J.-F. Chamberland, and K. R. Narayanan, “Unsourced random access with coded compressed sensing: Integrating AMP and belief propagation,” IEEE Trans. Inf. Theory, vol. 68, no. 4, pp. 2384–2409, Apr. 2022.

[13] J. Wang, Z. Zhang, X. Chen, C. Zhong, and L. Hanzo, “Unsourced massive random access scheme exploiting Reed–Muller sequences,” IEEE Trans. Commun., vol. 70, no. 2, pp. 1290–1303, Feb. 2022.

[14] S. K. Shokhiv, K. Andreev, A. Frolov, and Y. Poliansky, “Energy efficient random access for the quasi-static fading MAC,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), Jul. 2019, pp. 2768–2772.

[15] S. K. Shokhiv, K. Andreev, A. Frolov, and Y. Poliansky, “Short-packet low-power coded access for massive MAC,” in Proc. 53rd Asilomar Conf. Signals, Syst., Comput., Nov. 2019, pp. 827–832.

[16] A. Fengler, O. Musa, P. Jung, and G. Caire, “Pilot-based unsourced random access with a massive MIMO receiver, interference cancellation, and power control,” IEEE J. Sel. Areas Commun., vol. 40, no. 5, pp. 1522–1534, May 2022.

[17] K. Andreev, P. Rybin, and A. Frolov, “Reed–Solomon coded compressed sensing for the unsourced random access,” in Proc. 17th Int. Symp. Wireless Commun. Syst. (ISWCS), Sep. 2021, pp. 1–5.

[18] 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. 2947–2959, Jun. 2018.

[19] 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, Jun. 2018.

[20] J. Vila and P. Schniter, “Expectation-maximization Bernoulli–Gaussian approximate message passing,” in Proc. Conf. Rec. 45th Asilomar Conf. Signals, Syst., Comput. (ASILOMAR), Nov. 2011, pp. 799–803.

[21] J. Ma and L. Ping, “Orthogonal AMP,” IEEE Access, vol. 5, pp. 2020–2033, 2017.

[22] S. Haghjighatashoor, P. Jung, and G. Caire, “Improved scaling law for activity detection in massive MIMO systems,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2018, pp. 381–385.

[23] J. Daubechies, M. Defrise, and C. D. Mol, “An iterative thresholding algorithm for linear inverse problems with a sparsity constraint,” Commun. Pure Appl. Math., vol. 57, no. 11, pp. 1413–1457, Aug. 2004.
[24] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” SIAM J. Imag. Sci., vol. 2, no. 1, pp. 183–202, 2009.

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

[26] R. Kueng and P. Jung, “Robust nonnegative sparse recovery and the nullspace property of 0/1 measurements,” IEEE Trans. Inf. Theory, vol. 64, no. 2, pp. 689–703, Feb. 2018.

[27] J. Barbier and F. Krzakala, “Approximate message-passing decoder and capacity achieving sparse superposition codes,” IEEE Trans. Inf. Theory, vol. 63, no. 8, pp. 4894–4927, Aug. 2017.

[28] C.-K. Wen and K.-K. Wong, “Analysis of compressed sensing with jointly-sparse, overcomplete dictionaries,” 2014, arXiv:1402.3215.

[29] T. Hou, Y. Liu, T. Fu, and J. Barbier, “Sparse superposition codes under VAMP decoding with generic rotational invariant coding matrices,” 2022, arXiv:2202.04541.

[30] S. Foucart and H. Rauhut, A Mathematical Introduction to Compressive Sensing (Applied and Numerical Harmonic Analysis). Basel, Switzerland: Birkhäuser, 2013.

[31] T. Blumensath and M. E. Davies, “Iterative hard thresholding for compressed sensing,” Appl. Comput. Harmon. Anal., vol. 27, no. 3, pp. 265–274, Nov. 2009.

[32] M. Slawski and M. Hein, “Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization,” Electron. J. Statist., vol. 7, pp. 3004–3056, Jan. 2013.

[33] E. J. Candès and T. Tao, “Decoding by linear programming,” IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[34] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.

[35] J. Dong, J. Zhang, Y. Shi, and J. H. Wang, “Faster activity and data detection in massive random access: A multiarmed bandit approach,” IEEE Internet Things J., vol. 9, no. 15, pp. 13664–13678, Aug. 2022.

[36] M. Slawski and M. Hein, “Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization,” Electron. J. Statist., vol. 7, pp. 3004–3056, Jan. 2013.

[37] E. J. Candès and T. Tao, “Decoding by linear programming,” IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[38] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.

[39] J. Dong, J. Zhang, Y. Shi, and J. H. Wang, “Faster activity and data detection in massive random access: A multiarmed bandit approach,” IEEE Internet Things J., vol. 9, no. 15, pp. 13664–13678, Aug. 2022.

[40] M. Slawski and M. Hein, “Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization,” Electron. J. Statist., vol. 7, pp. 3004–3056, Jan. 2013.

[41] E. J. Candès and T. Tao, “Decoding by linear programming,” IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[42] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.

[43] J. Dong, J. Zhang, Y. Shi, and J. H. Wang, “Faster activity and data detection in massive random access: A multiarmed bandit approach,” IEEE Internet Things J., vol. 9, no. 15, pp. 13664–13678, Aug. 2022.

[44] M. Slawski and M. Hein, “Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization,” Electron. J. Statist., vol. 7, pp. 3004–3056, Jan. 2013.

[45] E. J. Candès and T. Tao, “Decoding by linear programming,” IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[46] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.

[47] J. Dong, J. Zhang, Y. Shi, and J. H. Wang, “Faster activity and data detection in massive random access: A multiarmed bandit approach,” IEEE Internet Things J., vol. 9, no. 15, pp. 13664–13678, Aug. 2022.

[48] M. Slawski and M. Hein, “Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization,” Electron. J. Statist., vol. 7, pp. 3004–3056, Jan. 2013.

[49] E. J. Candès and T. Tao, “Decoding by linear programming,” IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[50] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.

[51] J. Dong, J. Zhang, Y. Shi, and J. H. Wang, “Faster activity and data detection in massive random access: A multiarmed bandit approach,” IEEE Internet Things J., vol. 9, no. 15, pp. 13664–13678, Aug. 2022.

[52] M. Slawski and M. Hein, “Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization,” Electron. J. Statist., vol. 7, pp. 3004–3056, Jan. 2013.

[53] E. J. Candès and T. Tao, “Decoding by linear programming,” IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[54] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.

[55] J. Dong, J. Zhang, Y. Shi, and J. H. Wang, “Faster activity and data detection in massive random access: A multiarmed bandit approach,” IEEE Internet Things J., vol. 9, no. 15, pp. 13664–13678, Aug. 2022.

[56] M. Slawski and M. Hein, “Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization,” Electron. J. Statist., vol. 7, pp. 3004–3056, Jan. 2013.

[57] E. J. Candès and T. Tao, “Decoding by linear programming,” IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[58] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.

[59] J. Dong, J. Zhang, Y. Shi, and J. H. Wang, “Faster activity and data detection in massive random access: A multiarmed bandit approach,” IEEE Internet Things J., vol. 9, no. 15, pp. 13664–13678, Aug. 2022.

[60] M. Slawski and M. Hein, “Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization,” Electron. J. Statist., vol. 7, pp. 3004–3056, Jan. 2013.

[61] E. J. Candès and T. Tao, “Decoding by linear programming,” IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[62] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.