Proximal Gradient-Based Unfolding for Massive Random Access in IoT Networks

Yinan Zou, *Graduate Student Member, IEEE*, Yong Zhou, *Senior Member, IEEE*, Xu Chen, *Senior Member, IEEE*, and Yonina C. Eldar, *Fellow, IEEE*

***Abstract*—Grant-free random access is an effective technology for enabling low-overhead and low-latency massive access, where joint activity detection and channel estimation (JADCE) is a critical issue. Although existing compressed sensing algorithms can be applied for JADCE, they usually fail to simultaneously harvest the following properties: effective sparsity inducing, fast convergence, robust to different pilot sequences, and adaptive to time-varying networks. To this end, we propose an unfolding framework for JADCE based on the proximal gradient method. Specifically, we formulate the JADCE problem as a group-row-sparse matrix recovery problem and leverage a minimax concave penalty rather than the widely-used ℓ_1-norm to induce sparsity. We then develop a proximal gradient-based unfolding neural network that parameterizes the algorithmic iterations. To improve convergence rate, we incorporate momentum into the unfolding neural network, and prove the accelerated convergence theoretically. Based on the convergence analysis, we further develop an adaptive-tuning algorithm, which adjusts its parameters to different signal-to-noise ratio settings. Simulations show that the proposed unfolding neural network achieves better recovery performance, convergence rate, and adaptivity than current baselines.**

***Index Terms*—Massive random access, compressed sensing, proximal gradient unfolding, joint activity detection and channel estimation.**

I. Introduction

Massive machine-type communications (mMTC) is expected to connect a massive number of Internet of Things (IoT) devices [2]. Because of the sporadic short-packet communication and massive connectivity, adopting the conventional grant-based random access strategy to support mMTC may lead to overwhelming signaling overhead, thereby introducing significant access latency. Grant-free random access has received extensive attention, given its potential to enable low-latency and low-overhead massive access [3]. Specifically, without waiting for the grant, each IoT device directly transmits its data to the base station (BS) after sending a pilot sequence, which significantly reduces the signaling overhead. To fully exploit the advantages of grant-free random access, it is essential to achieve joint activity detection and channel estimation (JADCE) according to the pilot sequences received at the BS.

Because of the sporadic traffic of IoT devices and large antenna array at the BS, JADCE is usually modeled as different multiple measurement vector (MMV) compressed sensing (CS) problems [4], [5], [6] and then tackled by applying sparse signal processing methods. In particular, the JADCE problem can be formulated as group least absolute shrinkage and selection operator (LASSO), which can be solved by the iterative shrinkage thresholding algorithm (ISTA) [7], [8]. Apart from ISTA, other optimization-based algorithms [9], [10], [11], [12] have also been developed for JADCE. The authors in [13] proposed an approximate message passing (AMP)-based algorithm for JADCE in massive multiple-input multiple-output (MIMO) systems. AMP was further extended for activity detection in multi-cell networks [14]. In addition, the use of AMP for reconfigurable intelligent surface (RIS)-assisted massive access systems was studied in [15]. Despite the aforementioned studies, AMP-based algorithms may not converge in scenarios with either ill-conditioned or non-Gaussian pilot sequences [16], [17]. Moreover, optimization-based methods often have slow convergence and high computation complexity, and obtain sub-optimal solutions in practice, leading to non-negligible performance gap to the optimal solution.

Deep learning (DL) was emerged as a disruptive technique to tackle different optimization problems in wireless networks [18], including sparse signal recovery. In order to enable model-driven learning design for sparse signal recovery, unfolding iterative algorithms as recurrent neural networks (RNN) [19], [20] is an effective strategy. Different from the optimization-based methods that manually fix the parameters throughout the iterations, RNN adaptively tunes the parameters in each unfolding layer according to the training data, which accelerates convergence and leads to performance improvement. The authors in [21] and [22] proposed to unfold the generic ISTA and AMP into learned

Manuscript received 29 June 2023; revised 7 February 2024; accepted 10 June 2024. Date of publication 25 June 2024; date of current version 11 October 2024. This work was supported in part by the National Natural Science Foundation of China under Grant U20A20159, in part by the Natural Science Foundation of Shanghai under Grant 23ZR1442800, in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2023B1515120058 and Grant 2021B151520008, and in part by Guangzhou Basic and Applied Basic Research Program under Grant 2024A04J6367. An earlier version of this paper was presented in part at the IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, in 2021 [DOI: 10.1109/GLOBECOM46510.2021.9685447]. The associate editor coordinating the review of this article and approving it for publication was S. K. Mohammed. *(Corresponding author: Yong Zhou.)*

Yinan Zou and Yong Zhou are with the School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China (e-mail: yinan734@gmail.com; zhouyong@shanghaitech.edu.cn).

Xu Chen is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China (e-mail: chenxu35@mail.sysu.edu.cn).

Yonina C. Eldar is with the Faculty of Math and CS, Weizmann Institute of Science, Rehovot 7610001, Israel (e-mail: yonina.eldar@weizmann.ac.il).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2024.3416375.

Digital Object Identifier 10.1109/TWC.2024.3416375

ISTA (LISTA) and learned AMP (LAMP), respectively. The authors in [23] and [24] simplified the LISTA structure by studying its theoretical properties and proved its linear convergence. In [25], a LISTA framework was developed for group sparsity. To improve recovery performance, [26] considered an auto-encoder neural network to jointly design the pilot sequence matrix and recover sparse signal. By exploiting the domain knowledge and channel structure, the authors in [27] proposed DL-based approaches to aid the message passing algorithm. An asynchronous grant-free random access system was studied in [28], where different LAMP-based structures were designed to balance the tradeoff between performance and complexity. These studies [25], [26], [27] leveraged the widely-used ℓ_1-norm as the sparsity-inducing penalty (SIP).

To further promote sparse solutions, a proximal operator method was unfolded as an RNN for non-convex SIP-regularized problems in [29]. Though the scalar operator-based unfolding structure in [29] is effective for single measurement vector (SMV) problems, it does not consider the group-sparse structure that exists in the JADCE problem. Furthermore, these DL-based methods [25], [26], [27], [28], [29], [30] are developed based on a common assumption that the training and test datasets share the same distribution, i.e., signal-to-noise ratio (SNR) and device active ratio remain unchanged in the training and test stages. However, in many practical IoT networks, SNR and device active ratio are time-varying, which leads to a discrepancy between the training and test datasets. Hence, existing DL-based algorithms cannot be directly applied in such dynamic environments. An intuitive method to tackle this issue is to collect a new training dataset and re-train the neural network, which, however, incurs excessive communication and computation overhead for data collection and training. The authors in [31] proposed an adaptive scheme based on LISTA. However, how to develop an adaptive method for JADCE problems with group-sparse channel matrix and non-convex SIP has not been studied.

In this paper, we propose an adaptive unfolding neural network framework for JADCE based on a non-convex regularizer for group-sparsity, which ensures robustness to non-Gaussian pilot sequences, achieves fast convergence with theoretical guarantees, and adapts to time-varying device active ratio and SNR. The exist works [13], [25], [29], [31], however, only achieve some of these properties. As an effective approach to restrain oscillation and accelerate convergence, we incorporate momentum into the unfolding neural network. Though we do not analyze the spectral efficiency in this paper, our proposed unfolding framework significantly reduces the JADCE error and adapts to time-varying wireless networks, which facilitates the subsequent data transmission [32], [33], [34]. The main contributions of this paper are summarized as follows:

- We formulate the JADCE problem as a minimax concave penalty (MCP) regularized group-row-sparse matrix recovery problem. To efficiently solve this challenging problem, we propose a light-weight unfolding neural network, termed analytic learned proximal gradient method (ALPGM).
- To further improve convergence rate, we incorporate momentum into ALPGM and propose an accelerated variant of ALPGM, termed ALPGM with momentum (ALPGM-MM). Theoretical analysis is conducted to characterize the convergence of ALPGM-MM. The theoretical result shows that ALPGM-MM has the no-false-positive property and enjoys a better convergence rate than analytic LSITA for group sparsity (ALISTA-GS) in [25] under certain parameter settings.
- Based on the convergence analysis, we further propose an adaptive-tuning scheme, termed learned proximal gradient method with adaptive-tuning parameters (LPGM-AT), which adapts to the variation of the device active ratio and SNR. The hyperparameters in LPGM-AT are optimized by grid search rather than back-propagation, which significantly reduces the computational complexity. The proposed LPGM-AT adaptively adjusts the network parameters according to the input data, and hence facilitates JADCE in time-varying IoT networks.
- Simulations show that the proposed ALPGM and ALPGM-MM achieve better recovery performance than the baselines. Moreover, benefiting from the momentum acceleration, the proposed ALPGM-MM exhibits faster convergence rate than ALPGM. LPGM-AT significantly outperforms ALPGM and ALPGM-MM on the test dataset that differs from the training dataset in terms of device active ratio and SNR.

The remainder of this paper is organized as follows. System model and problem formulation are described in Section II. In Section III, we propose three unfolding neural networks for tackling the JADCE problem. We present simulation results in Section IV. Finally, the paper is concluded in Section V.

Notations: We denote $[N] = [1, \ldots, N]$. We use $\mathbb{R}^N$ and $\mathbb{C}^N$ to denote the real and complex domains of dimension N, respectively, $|S|$ denotes the cardinality of set S and $\text{supp}(\boldsymbol{x})$ is the support of vector $\boldsymbol{x} = [x_1, \ldots, x_N] \in \mathbb{R}^N$. We denote the sign function and the generalized inverse of a matrix as $\text{sign}(\cdot)$ and $\boldsymbol{X}^\dagger$, respectively. For matrix $\boldsymbol{X} \in \mathbb{R}^{N\times M}$ and index sets $\mathcal{I} = \{i_1, \ldots, i_n\} \subset [N]$, $\mathcal{J} = \{j_1, \ldots, j_m\} \subset [M]$, we denote $\boldsymbol{X}_{\mathcal{I},\mathcal{J}}$ as the (sub)matrix that includes the entries from the rows of $\boldsymbol{X}$ indicated by $\mathcal{I}$ and the columns indicated by $\mathcal{J}$. When $\mathcal{I} = [N]$ (or $\mathcal{J} = [M]$), we denote $\boldsymbol{X}_{:,\mathcal{J}}$ (or $\boldsymbol{X}_{\mathcal{I},:}$) as the submatrix that contains the entries from all the rows (or all the columns) of $\boldsymbol{X}$ specified by the index set.

II. System Model and Problem Formulation

A. System Model

In this paper, we consider a single-cell IoT network, which consists of N single-antenna IoT devices and one M-antenna BS. Compared to the number of BS antennas, the number of IoT devices is generally much larger, i.e., $N \gg M$. According to the principle of grant-free random access, each IoT device with sporadic traffic independently makes the transmission decision, and a small number of IoT devices decide to transmit in each transmission block. Specifically, the active devices, without the need to obtain a scheduling grant from the BS, send their pre-allocated pilot sequences along

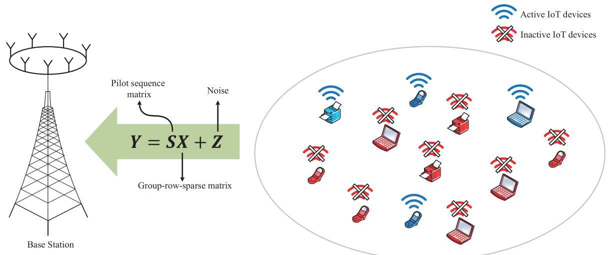


Fig. 1. An illustration of an IoT network that consists of massive devices with sporadic traffic.

with their short-length data, while the inactive devices keep silent. In any transmission block, we denote $a_n = 1$ if device n is active, and $a_n = 0$ otherwise. The uplink channel response between IoT device n and the BS is denoted as $\boldsymbol{h}_n \in \mathbb{C}^M$, which remains unchanged in each transmission block and varies independently across different blocks and devices [35]. With synchronized pilot transmissions from active devices, the signal $\boldsymbol{y}(\ell) \in \mathbb{C}^M$ received at the BS is

$$\boldsymbol{y}(\ell) = \sum_{n=1}^{N} \boldsymbol{h}_n a_n s_n(\ell) + \boldsymbol{z}(\ell), \quad \ell = 1, \ldots, L, \tag{1}$$

where $s_n(\ell)$ is the ℓ-th pilot symbol transmitted by device n, L denotes the pilot length, and $\boldsymbol{z}(\ell) \in \mathbb{C}^M$ denotes the additive white Gaussian noise (AWGN) vector with each entry following distribution $\mathcal{CN}(0, \sigma^2)$. Compared to the device number, the pilot sequence length is generally much smaller, i.e., $L \ll N$, which makes it impractical for all devices to have orthogonal sequences. As a result, each device is assigned a non-orthogonal but unique sequence.

By denoting $\boldsymbol{Y} = [\boldsymbol{y}(1), \ldots, \boldsymbol{y}(L)]^{\mathrm{T}} \in \mathbb{C}^{L \times M}$, $\boldsymbol{A} = \mathrm{Diag}(a_1, \ldots, a_N) \in \mathbb{R}^{N \times N}$, $\boldsymbol{S} = [\boldsymbol{s}(1), \ldots, \boldsymbol{s}(L)]^{\mathrm{T}} \in \mathbb{C}^{L \times N}$ with $\boldsymbol{s}(\ell) = [s_1(\ell), \ldots, s_N(\ell)]^{\mathrm{T}} \in \mathbb{C}^N$, $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_N]^{\mathrm{T}} \in \mathbb{C}^{N \times M}$, and $\boldsymbol{Z} = [\boldsymbol{z}(1), \ldots, \boldsymbol{z}(L)]^{\mathrm{T}} \in \mathbb{C}^{L \times M}$, the received signal at the BS is rewritten in matrix form as

$$\boldsymbol{Y} = \boldsymbol{SAH} + \boldsymbol{Z}. \tag{2}$$

Before decoding data, the BS conducts JADCE (i.e., recovering matrices $\boldsymbol{A}$ and $\boldsymbol{H}$) based on the received pilot signals. Denoting $\boldsymbol{X} = \boldsymbol{AH} \in \mathbb{C}^{N \times M}$, we rewrite (2) as

$$\boldsymbol{Y} = \boldsymbol{SX} + \boldsymbol{Z}. \tag{3}$$

B. Problem Formulation

Since the device activity matrix $\boldsymbol{A}$ is diagonal, we have $\boldsymbol{X} = [a_1\boldsymbol{h}_1, \ldots, a_N\boldsymbol{h}_N]^{\mathrm{T}}$. If device n is inactive, then all entries of the n-th row of matrix $\boldsymbol{X}$ are zero. Thus, matrix $\boldsymbol{X}$ has the structure of group-sparsity in rows and all columns share the same support. Achieving JADCE is equivalent to recovering the row support of $\boldsymbol{X}$ and the elements of nonzero rows based on the noisy observation $\boldsymbol{Y}$ at the BS. Such a matrix recovery problem is given by

$$\mathcal{P} : \underset{\boldsymbol{X} \in \mathbb{C}^{N \times M}}{\text{minimize}} \ \frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{SX}\|_F^2 + \lambda G(\boldsymbol{X}), \tag{4}$$

where $\lambda > 0$ is the regularization parameter, and $G(\boldsymbol{X})$ is an SIP term introduced to induce the group-row-sparsity of matrix $\boldsymbol{X}$.

In the following, (3) is rewritten as its real-valued counterpart

$$\tilde{\boldsymbol{Y}} = \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}} + \tilde{\boldsymbol{Z}} = \begin{bmatrix} \mathcal{R}\{\boldsymbol{S}\} & -\mathcal{I}\{\boldsymbol{S}\} \\ \mathcal{I}\{\boldsymbol{S}\} & \mathcal{R}\{\boldsymbol{S}\} \end{bmatrix} \begin{bmatrix} \mathcal{R}\{\boldsymbol{X}\} \\ \mathcal{I}\{\boldsymbol{X}\} \end{bmatrix} + \begin{bmatrix} \mathcal{R}\{\boldsymbol{Z}\} \\ \mathcal{I}\{\boldsymbol{Z}\} \end{bmatrix}, \tag{5}$$

where $\mathcal{R}\{\cdot\}$ and $\mathcal{I}\{\cdot\}$ denote the real and imaginary parts of a complex matrix. Hence, problem $\mathcal{P}$ is rewritten as

$$\mathcal{P}_r : \underset{\tilde{\boldsymbol{X}} \in \mathbb{R}^{2N \times M}}{\text{minimize}} \ \frac{1}{2} \|\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}\|_F^2 + \lambda G(\tilde{\boldsymbol{X}}). \tag{6}$$

To induce a group-sparse solution, the authors in [25] and [36] adopted a convex SIP in the form of $G(\tilde{\boldsymbol{X}}) = \sum_{i=1}^{2N} \|\tilde{\boldsymbol{X}}_{i,:}\|_2$ (i.e., mixed ℓ_1/ℓ_2-norm), and reformulated problem $\mathcal{P}_r$ as group LASSO [8]. Since MCP [37] induces further sparsity than the ℓ_1-norm, we choose MCP as the SIP and rewrite problem $\mathcal{P}_r$ as the following group MCP problem [38]

$$\text{Group MCP} : \underset{\tilde{\boldsymbol{X}} \in \mathbb{R}^{2N \times M}}{\text{minimize}} \ \frac{1}{2} \|\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}\|_F^2 + \lambda \sum_{i=1}^{2N} g_\eta(\|\tilde{\boldsymbol{X}}_{i,:}\|_2), \tag{7}$$

where

$$g_\eta(z) = \begin{cases} |z| - \eta z^2, & \text{if } |z| \le \dfrac{1}{2\eta}, \\ \dfrac{1}{4\eta}, & \text{if } |z| > \dfrac{1}{2\eta}. \end{cases} \tag{8}$$

C. Conventional Proximal Gradient Method

For group MCP, we apply the following iterative proximal gradient method (PGM) to recover real-valued matrix $\tilde{\boldsymbol{X}}$

$$\tilde{\boldsymbol{X}}^{k+1} = P_{\lambda\gamma_k, f_{\eta_k}}\left(\tilde{\boldsymbol{X}}^k + \gamma_k \tilde{\boldsymbol{S}}^{\mathrm{T}}(\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^k)\right), \tag{9}$$

where γ_k denotes the step-size and $\tilde{\boldsymbol{X}}^k$ is an estimation of $\tilde{\boldsymbol{X}}$ at iteration k. The iterative proximal gradient method (9) originates from [38], which first extends nonconvex penalty MCP to a group selection problem, and then modifies ISTA to fit group MCP by replacing the soft-thresholding operator with the proximal operator with MCP.

The multivariate proximal operator $P_{\lambda\gamma_k, f_{\eta_k}}(\cdot)$ is given by

$$P_{\theta_k, f_{\eta_k}}(\tilde{\boldsymbol{X}}_{i,:}) = \arg\min_{\tilde{\boldsymbol{U}}_{i,:}} \frac{1}{2}\|\tilde{\boldsymbol{U}}_{i,:} - \tilde{\boldsymbol{X}}_{i,:}\|_2^2 + f_{\eta_k}(\tilde{\boldsymbol{U}}_{i,:}), \tag{10}$$

with $f_{\eta_k}(\tilde{\boldsymbol{U}}_{i,:}) = \theta_k g_{\eta_k}(\|\tilde{\boldsymbol{U}}_{i,:}\|_2)$ and $\theta_k = \lambda\gamma_k$. The univariate proximal operator can be written as $\hat{P}_{\theta_k, f_{\eta_k}}(x) = \arg\min_u \frac{1}{2}(u - x)^2 + \hat{f}_{\eta_k}(u)$, where $\hat{f}_{\eta_k}(u) = \theta_k g_{\eta_k}(u)$ [39]. To have a well-defined minimum, we should have $\eta_k < \frac{1}{2\theta_k}$, which yields

$$\hat{P}_{\theta_k, f_{\eta_k}}(x) = \begin{cases} 0, & \text{if } |x| \le \theta_k, \\ \dfrac{x - \theta_k \mathrm{sign}(x)}{1 - 2\theta_k\eta_k}, & \text{if } \theta_k < |x| \le \dfrac{1}{2\eta_k}, \\ x, & \text{if } |x| > \dfrac{1}{2\eta_k}. \end{cases} \tag{11}$$

Based on [40, Theorem 6.18] and (11), we obtain

$$P_{\theta_k, f_{\eta_k}}(\tilde{\boldsymbol{X}}_{i,:}) = \begin{cases} \hat{P}_{\theta_k, f_{\eta_k}}(\|\tilde{\boldsymbol{X}}_{i,:}\|_2)\dfrac{\tilde{\boldsymbol{X}}_{i,:}}{\|\tilde{\boldsymbol{X}}_{i,:}\|_2}, & \text{if } \tilde{\boldsymbol{X}}_{i,:} \neq \boldsymbol{0}, \\ \boldsymbol{0}, & \text{otherwise.} \end{cases} \tag{12}$$

The resulting PGM can solve problem (7) [41]. However, it has several limitations. First, PGM achieves sublinear convergence rate and usually takes many iterations to converge. In time-varying IoT networks, the variations of device active ratio and SNR cause PGM to re-execute, which incurs a high computational complexity. Second, an inappropriate choice of the regularization parameter λ may severely degrade performance of PGM. Third, the values of the step-size γ_k and parameter η_k influence the convergence rate, and are generally tricky to choose. To tackle these limitations, we propose an unfolding neural network framework to improve recovery performance and accelerate the convergence by learning key parameters λ, γ_k, and η_k.

III. Proposed Unfolding Framework

This section proposes an unfolding framework that tackles the matrix recovery problem by unfolding the conventional PGM discussed in Section II-C.

A. ALPGM

Following the idea of algorithm unfolding, we unfold the iteration in (9) as an RNN. By treating $\tilde{\boldsymbol{X}}^k$ and $\tilde{\boldsymbol{X}}^{k+1}$ as the input and output of the activation function $P_{\lambda\gamma_k, f_{\eta_k}}(\cdot)$, respectively, (9) can be mapped to a one-layer neural network. Therefore, the K iterations are implemented by a K-layer RNN, where each neural network layer corresponds to a specific iteration of PGM. As using a measurement matrix with lower coherence yields a better performance in recovering sparse signals, we solve the following optimization problem to obtain matrix $\boldsymbol{B}^{\mathrm{T}}$ that has a small 'generalized coherence' [24] with $\tilde{\boldsymbol{S}}$

$$\underset{\boldsymbol{B}\in\mathbb{R}^{2N\times 2L}}{\text{minimize}} \quad \|\boldsymbol{B}\tilde{\boldsymbol{S}}\|_F^2 \tag{13}$$

$$\text{subject to} \quad \boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,i} = 1, \ \forall i \in [2N]. \tag{14}$$

We utilize the projected gradient descent (PGD) method to solve problem (13) [24]. We plot the histograms of off-diagonal elements of matrices $\boldsymbol{B}\tilde{\boldsymbol{S}}$ and $\tilde{\boldsymbol{S}}^{\mathrm{T}}\tilde{\boldsymbol{S}}$ in Fig. 2. It can be observed that the maximum absolute value of the off-diagonal elements of matrix $\boldsymbol{B}\tilde{\boldsymbol{S}}$ is smaller than that of matrix $\tilde{\boldsymbol{S}}^{\mathrm{T}}\tilde{\boldsymbol{S}}$, which indicates that the 'generalized coherence' between $\boldsymbol{B}^{\mathrm{T}}$ and $\tilde{\boldsymbol{S}}$ is smaller than the mutual coherence of $\tilde{\boldsymbol{S}}$. We replace $\tilde{\boldsymbol{S}}^{\mathrm{T}}$ by matrix $\boldsymbol{B}$, and obtain the unfolding neural network termed ALPGM

$$\tilde{\boldsymbol{X}}^{k+1} = P_{\theta_k, f_{\eta_k}}\left(\tilde{\boldsymbol{X}}^k + \gamma_k \boldsymbol{B}(\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^k)\right), k = 0, \ldots, K-1, \tag{15}$$

where $\theta_k = \lambda\gamma_k$ is the thresholding parameter of layer k. The trainable parameters are $\boldsymbol{\Theta} = \{\gamma_k, \theta_k, \eta_k\}_{k=0}^{K-1}$. The proposed ALPGM is shown in Fig. 3.

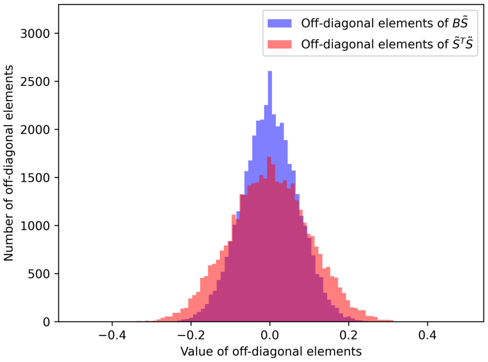


Fig. 2. Histograms of off-diagonal elements of matrices $\boldsymbol{B}\tilde{\boldsymbol{S}}$ and $\tilde{\boldsymbol{S}}^{\mathrm{T}}\tilde{\boldsymbol{S}}$ when $\boldsymbol{S}$ is a complex Gaussian matrix with $L = 50$ and $N = 100$.

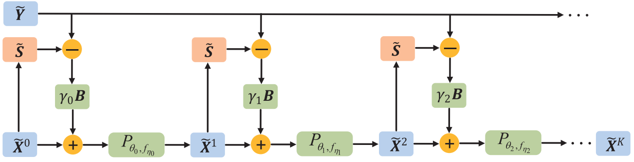


Fig. 3. An illustration of the proposed ALPGM, where $\{\gamma_k, \theta_k, \eta_k\}_{k=0}^{K-1}$ are trainable parameters.

We note that an intuitive method to unfold (9) is to fix $\tilde{\boldsymbol{S}}^{\mathrm{T}}$ and then directly learn $\{\gamma_k, \theta_k, \eta_k\}$. Another method is to replace $\gamma_k\tilde{\boldsymbol{S}}^{\mathrm{T}}$ by $\boldsymbol{B}^k$ and then learn $\{\boldsymbol{B}^k, \theta_k, \eta_k\}$. These two methods achieve poorer recovery performance than our proposed ALPGM, as will be shown in Section IV-A.

B. ALPGM-MM

For vanilla gradient descent, the gradient may not always point towards the minimum, which results in an oscillating update path and slow convergence. One solution is to utilize momentum to mitigate oscillations and speed up convergence [42]. Hence, we propose ALGPM-MM where we introduce a momentum term relating to $\tilde{\boldsymbol{X}}^{k-1}$ into the update of $\tilde{\boldsymbol{X}}^{k+1}$ in ALGPM, i.e.,

$$\tilde{\boldsymbol{X}}^{k+1} = \begin{cases} P_{\theta_k, f_{\eta_k}}\left(\tilde{\boldsymbol{X}}^k + \gamma_k\boldsymbol{B}(\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^k)\right), \\ \qquad\qquad \text{if } k = 0, \\ P_{\theta_k, f_{\eta_k}}\left(\tilde{\boldsymbol{X}}^k + \gamma_k\boldsymbol{B}(\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^k) + \beta_k(\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^{k-1})\right), \\ \qquad\qquad \text{if } k = 1, \ldots, K-1, \end{cases} \tag{16}$$

where β_k is the momentum parameter. The trainable parameters are $\{\gamma_k, \theta_k, \eta_k\}_{k=0}^{K-1}$ and $\{\beta_k\}_{k=1}^{K-1}$. The proposed ALPGM-MM is shown in Fig. 4.

Since the update of $\tilde{\boldsymbol{X}}^{k+1}$ is dependent upon $\tilde{\boldsymbol{X}}^k$ and $\tilde{\boldsymbol{X}}^{k-1}$, it is difficult to directly analyze the convergence of ALPGM-MM. Besides, the multivariate proximal operator with respect to MCP also brings a critical challenge for convergence analysis. For tractability of the convergence analysis of ALPGM-MM, the following problem replaces (13) to ensure that the matrix $\boldsymbol{B}\tilde{\boldsymbol{S}}$ is symmetric. By defining $\boldsymbol{B} = ((\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G})\tilde{\boldsymbol{S}})^{\mathrm{T}} \in \mathbb{R}^{2N\times 2L}$ with $\boldsymbol{G} \in \mathbb{R}^{2L\times 2L}$, problem (13) can

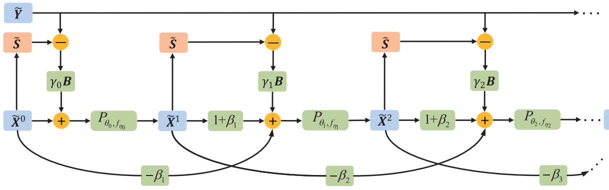


Fig. 4. An illustration of the proposed ALPGM-MM, where $\{\gamma_k, \theta_k, \eta_k\}_{k=0}^{K-1}$ and $\{\beta_k\}_{k=1}^{K-1}$ are trainable parameters.

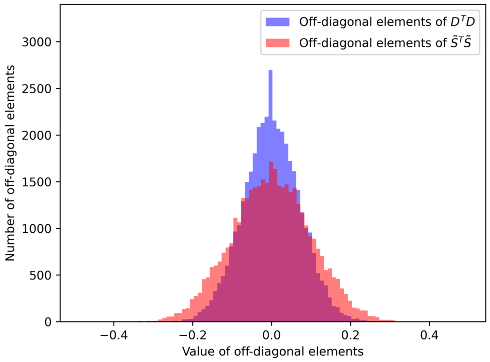


Fig. 5. Histograms of off-diagonal elements of matrices $\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D}$ and $\tilde{\boldsymbol{S}}^{\mathrm{T}}\tilde{\boldsymbol{S}}$ when $\boldsymbol{S}$ is a complex Gaussian matrix with $L = 50$ and $N = 100$.

be written as

$$\underset{\boldsymbol{G}\in\mathbb{R}^{2L\times 2L}}{\text{minimize}} \quad \|\tilde{\boldsymbol{S}}^{\mathrm{T}}\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G}\tilde{\boldsymbol{S}} - \boldsymbol{I}\|_F^2 \tag{17}$$

$$\text{subject to} \quad (\tilde{\boldsymbol{S}}^{\mathrm{T}}\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G}\tilde{\boldsymbol{S}})_{i,i} = 1, \forall i \in [2N]. \tag{18}$$

However, constraint (18) is difficult to be tackled. $\boldsymbol{G}$ and $\tilde{\boldsymbol{S}}$ are closely coupled in constraint (18), i.e., matrix $\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G}$ is surrounded by matrices $\tilde{\boldsymbol{S}}^{\mathrm{T}}$ and $\tilde{\boldsymbol{S}}$. It is hard to directly use projected gradient descent to design a projection operator to satisfy constraint $(\tilde{\boldsymbol{S}}^{\mathrm{T}}\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G}\tilde{\boldsymbol{S}})_{i,i} = 1$. Hence, we define an auxiliary matrix $\boldsymbol{D} = \boldsymbol{G}\tilde{\boldsymbol{S}} \in \mathbb{R}^{2L\times 2N}$ and reformulate problem (17) as

$$\underset{\substack{\boldsymbol{G}\in\mathbb{R}^{2L\times 2L},\\ \boldsymbol{D}\in\mathbb{R}^{2L\times 2N}}}{\text{minimize}} \quad \|\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D} - \boldsymbol{I}\|_F^2 \tag{19}$$

$$\text{subject to} \quad (\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D})_{i,i} = 1, \forall i \in [2N], \tag{20}$$

$$\boldsymbol{D} = \boldsymbol{G}\tilde{\boldsymbol{S}}. \tag{21}$$

Then, we convert the equality constraint $\boldsymbol{D} = \boldsymbol{G}\tilde{\boldsymbol{S}}$ into penalty term $\|\boldsymbol{D} - \boldsymbol{G}\tilde{\boldsymbol{S}}\|_F^2$ in the objective function. Therefore, we reformulate problem (19) as

$$\underset{\substack{\boldsymbol{G}\in\mathbb{R}^{2L\times 2L},\\ \boldsymbol{D}\in\mathbb{R}^{2L\times 2N}}}{\text{minimize}} \quad \|\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D} - \boldsymbol{I}\|_F^2 + \tau\|\boldsymbol{D} - \boldsymbol{G}\tilde{\boldsymbol{S}}\|_F^2 \tag{22}$$

$$\text{subject to} \quad (\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D})_{i,i} = 1, \forall i \in [2N], \tag{23}$$

where $\tau > 0$ denotes the regularization parameter. We can also adopt the PGD method to solve this problem [31], [43]. According to theorem on convergence of penalty method [44, Chapter 13], as the regularization parameter approaches infinity, the solution of the penalty problem converges to the solution of the original problem.

We plot the histograms of off-diagonal elements of matrices $\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D}$ and $\tilde{\boldsymbol{S}}^{\mathrm{T}}\tilde{\boldsymbol{S}}$. As shown in Fig. 5, the maximum absolute value of the off-diagonal elements of $\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D}$ is smaller than that of $\tilde{\boldsymbol{S}}^{\mathrm{T}}\tilde{\boldsymbol{S}}$, which indicates that the mutual coherence of matrix $\boldsymbol{D}$ is smaller than the mutual coherence of $\tilde{\boldsymbol{S}}$ and thus matrix $\boldsymbol{D}$ satisfies the coherence property. Through the above reformulation, matrix $\boldsymbol{B}\tilde{\boldsymbol{S}}$ is guaranteed to be a positive semidefinite matrix. In summary, before the training phase of ALPGM-MM, we solve problem (22) to obtain $\boldsymbol{G}$, and then obtain $\boldsymbol{B} = ((\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G})\tilde{\boldsymbol{S}})^{\mathrm{T}}$.

In Theorem 1, we show that ALPGM-MM has the no-false-positive property and achieves a faster convergence rate than ALISTA-GS in [25]. We denote $\psi(\tilde{\boldsymbol{X}}) = [\|\tilde{\boldsymbol{X}}_{1,:}\|_2, \ldots, \|\tilde{\boldsymbol{X}}_{2N,:}\|_2]^{\mathrm{T}}$ and define the mutual coherence of $\boldsymbol{D}$ as $\phi \triangleq \max_{i\neq j} |\boldsymbol{D}_{:,i}^{\mathrm{T}}\boldsymbol{D}_{:,j}|$. As in [23], [24], [25], [31], and [29], signal $\tilde{\boldsymbol{X}}^*$ and noise $\tilde{\boldsymbol{Z}}$ are assumed to belong to the set $\mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon) \triangleq \{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \,|\, 0 < \underline{\mu_x} \le \|\tilde{\boldsymbol{X}}_{i,:}^*\|_2 \le \mu_x, \forall i \in S, |S| \le s, \|\tilde{\boldsymbol{Z}}\|_F \le \epsilon\}$, where $\mathrm{supp}(\psi(\tilde{\boldsymbol{X}}^*))$ is denoted as S.

Theorem 1: *For ALPGM-MM, we denote the input as* $\tilde{\boldsymbol{Y}} = \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^* + \tilde{\boldsymbol{Z}}$ *and* $\tilde{\boldsymbol{X}}^0 = \boldsymbol{0}$, *the output as* $\{\tilde{\boldsymbol{X}}^k\}_{k=1}^{\infty}$, *and* $\|\boldsymbol{X}\|_{2,1} = \sum_n \|\boldsymbol{X}_{n,:}\|_2$. *If* $c_{\phi s} \triangleq (2s-1)\phi < 1$, $\|\boldsymbol{B}\|_{2,1} \le \mu_B$, *and the parameters* $\{\theta_k, \eta_k, \gamma_k, \beta_k\}$ *satisfy*

$$\phi \sup_{\substack{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}})\in\\ \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)}} \|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_{2,1} + \mu_B\epsilon = \theta_k \le \frac{1}{2\eta_k}, \quad \forall k, \tag{24}$$

$$\gamma_k = 1, \quad \forall k, \tag{25}$$

$$\beta_k \to \frac{1}{2s}\left(1 - \sqrt{1 - c_{\phi s}}\right)^2, \quad \textit{as } k \to \infty, \tag{26}$$

$$\eta_k \to \frac{1}{2\theta_k}, \quad \textit{as } k \to \infty, \tag{27}$$

then the sequence of iterations in (16) satisfies

$$\textit{supp}(\psi(\tilde{\boldsymbol{X}}^k)) \subseteq S, \forall k, \tag{28}$$

and

$$\|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_F \le C_0 \prod_{t=1}^{k} c^t + \frac{(1+s)\mu_B\epsilon}{1 - c_{\phi s}}, \forall k, \tag{29}$$

where $C_0 > 0$ *is a constant and* c^k *satisfies*

$$0 < c^k \le c_{\phi s} < 1, \forall k, \tag{30}$$

$$0 < c^k \le 1 - \sqrt{1 - c_{\phi s}}, \forall k > \left\lceil \frac{\log(\underline{\mu_x}) - \log(6C_0)}{\log(c_{\phi s})} \right\rceil + 2. \tag{31}$$

Proof: See Appendix A. □

According to Theorem 1, as long as the parameters satisfy (24)-(27), the no-false-positive property is satisfied, meaning that the index set of the rows containing non-zero elements of $\psi(\tilde{\boldsymbol{X}}^k)$ belongs to that of the ground truth. In other words, if the device is actually inactive, then it will not be detected as active. Based on the no-false-positive property, we prove that $\tilde{\boldsymbol{X}}^k$ converges to the vicinity of the ground truth $\tilde{\boldsymbol{X}}^*$, i.e., $\tilde{\boldsymbol{X}}^k$ is close to $\tilde{\boldsymbol{X}}^*$, implying that the JADCE error can be fairly small. Theorem 1 also demonstrates that ALPGM-MM achieves a linear convergence rate in a noisy scenario. Although the convergence rate is linear, the convergence rate of ALPGM-MM (i.e., $c_{\phi s}$) is better than the convergence rate in [1] (i.e., $\phi s(\sqrt{M}+1)-\phi$). This is because the multivariate proximal operator exploits the group-row-sparsity property, which accelerates the convergence. In addition, since $1-\sqrt{1-c_{\phi s}} < c_{\phi s}$ when $c_{\phi s} < 1$, the convergence performance of ALPGM-MM is better than that of ALISTA-GS in [25] under the same setting because the momentum term provides convergent acceleration.

As ALPGM shares a similar network with ALPGM-MM except the momentum part, the convergence analysis of ALPGM-MM can be reduced to that of ALPGM by removing the momentum part. Through some modifications of the proof of Theorem 1, one prove that ALPGM also achieves linear convergence rate.

From a theoretical perspective, Theorem 1 verifies the validity of ALPGM-MM under certain conditions of parameters $\{\theta_k, \eta_k, \gamma_k, \beta_k\}$. This assumption is only made for tractability of the analysis. Although the parameters learned by back-propagation may not necessarily satisfy the conditions in Theorem 1, ALPGM-MM with learned parameters still exhibits excellent performance and fast convergence in practice, as we show in Section IV-B.

C. LPGM-AT

Most DL-based approaches including the proposed ALPGM and ALPGM-MM rely on the assumption that the SNR and the device active ratio remain the same during the training and test stages. As a result, they may not work well in dynamic IoT networks, where the SNR and device active ratio are time-varying. To tackle this problem, we further develop an adaptive-tuning algorithm, termed LPGM-AT, for dynamic IoT networks.

In ALPGM-MM, $\{\gamma_k, \theta_k, \eta_k, \beta_k\}$ are regarded as the trainable parameters, and optimized by back-propagation on the training dataset. Thus, the optimized parameters entirely depend on training data and are applicable for test data that follows the same distribution as training data. The drawback is that a minor discrepancy between training and test data distributions may incur severe performance degradation. To address this issue and achieve algorithmic robustness, we turn our attention to optimize the parameters according to $\tilde{\boldsymbol{X}}^k$ and $\tilde{\boldsymbol{Y}}$.

Since $\tilde{\boldsymbol{Y}} = \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^* + \tilde{\boldsymbol{Z}}$, we obtain $\tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{Y}} = \tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^* + \tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{Z}}$, where $\tilde{\boldsymbol{S}}^\dagger$ is the generalized inverse of $\tilde{\boldsymbol{S}}$. Through adding $\tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^k$ and taking the norm on both sides, we obtain $\|\tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{Y}}\|_{2,1} = \|\tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{S}}(\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*) - \tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{Z}}\|_{2,1}$. We use $\|\tilde{\boldsymbol{S}}^\dagger(\tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{Y}})\|_{2,1}$ to approximate (24) because $\|\tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{S}}(\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*) - \tilde{\boldsymbol{S}}^\dagger\tilde{\boldsymbol{Z}}\|_{2,1} \approx \mathcal{O}(\|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_{2,1}) + \mathcal{O}(\epsilon)$. Therefore, the adaptive-tuning of thresholding parameter θ_k is designed as follows

$$\theta_k = c_\theta\|\tilde{\boldsymbol{S}}^\dagger(\tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{Y}})\|_{2,1}, \quad k = 0, \dots, K-1, \tag{32}$$

where $c_\theta > 0$ is a tunable hyperparameter. According to (32), we observe that the thresholding parameter θ_k only depends on $\tilde{\boldsymbol{X}}^k$ and $\boldsymbol{Y}$, and does not need the prior knowledge of $\tilde{\boldsymbol{X}}^*$.

According to (25), we set step-size parameter γ_k as

$$\gamma_k = 1, \quad k = 0, \dots, K-1. \tag{33}$$

In (26), the momentum parameter β_k approaches $\frac{1}{2s}(1-\sqrt{1-c_{\phi s}})^2$ with $c_{\phi s} = (2s-1)\phi$ when k approaches infinity. Note that $\frac{1}{2s}(1-\sqrt{1-c_{\phi s}})^2$ is a monotonic increasing function of s when $s > 1$. By following the same idea of getting rid of the dependence on $\tilde{\boldsymbol{X}}^*$, we utilize $\|\psi(\tilde{\boldsymbol{X}}^k)\|_0$ to approximate $\frac{1}{2s}(1-\sqrt{1-c_{\phi s}})^2$, because $\tilde{\boldsymbol{X}}^k$ converges to $\tilde{\boldsymbol{X}}^*$ while $\|\psi(\tilde{\boldsymbol{X}}^k)\|_0$ approaches $\|\psi(\tilde{\boldsymbol{X}}^*)\|_0$. Therefore, the adaptive-tuning of momentum parameter β_k is designed as follows

$$\beta_k = c_\beta\|\psi(\tilde{\boldsymbol{X}}^k)\|_0, \quad k = 1, \dots, K-1, \tag{34}$$

where $c_\beta > 0$ is a tunable hyperparameter.

By considering the coupling relationship between η_k and θ_k (i.e., $2\theta_k\eta_k < 1$), the adaptive-tuning of parameter η_k is designed as follows

$$\eta_k = \frac{1}{c_\eta\|\psi(\tilde{\boldsymbol{X}}^k)\|_0\theta_k}, \quad k = 0, \dots, K-1, \tag{35}$$

where $c_\eta > 0$ is a tunable hyperparameter.

We use grid search to find the best hyperparameters (i.e., c_θ, c_β, and c_η) instead of back-propagation in the training phase. Specifically, we execute the algorithm on the training dataset with a series of hyperparameter combinations and choose the hyperparameter combination that achieves the best performance. LPGM-AT only needs to optimize three hyperparameters, which significantly reduces the training complexity. Although DL can also be leveraged for optimizing the three hyperparameters, it entails a much higher computational complexity than grid search.

The values of hyperparameters $\{c_\theta, c_\beta, c_\eta\}$ are determined in the training phase. Once the training phase ends, the hyperparameters are fixed, and directly applied to the test datasets. According to (32)-(35), parameters $\{\theta_k, \beta_k, \eta_k\}$ rely on hyperparameters $\{c_\theta, c_\beta, c_\eta\}$, $\tilde{\boldsymbol{X}}^k$, and $\tilde{\boldsymbol{Y}}$. As the hyperparameters are fixed in the test phase, parameters $\{\theta_k, \beta_k, \eta_k\}$ only depend on $\tilde{\boldsymbol{X}}^k$ and $\tilde{\boldsymbol{Y}}$. For different distributions of the test dataset, parameters $\{\theta_k, \beta_k, \eta_k\}$ vary with $\tilde{\boldsymbol{X}}^k$ and $\tilde{\boldsymbol{Y}}$. Thus, our proposed LPGM-AT is self-adaptive for different test datasets. If the test dataset shares the same distribution with the training dataset, LPGM-AT achieves the same performance on both datasets. If the distribution of the test dataset differs from that of the training dataset, then LPGM-AT adapts to the unknown distribution of the test dataset.

D. Training and Testing Strategies

1) ALPGM and ALPGM-MM: For these two neural networks, we adopt supervised learning based on training set $\{\tilde{\boldsymbol{X}}_i^*, \tilde{\boldsymbol{Y}}_i\}_{i=1}^T$, where $\tilde{\boldsymbol{Y}}_i$ is the data, $\tilde{\boldsymbol{X}}_i^*$ is the corresponding label, and T is the size of the training set. We denote the output of K-layer RNN as $\tilde{\boldsymbol{X}}^K(\boldsymbol{\Theta}, \tilde{\boldsymbol{Y}}_i, \tilde{\boldsymbol{X}}^0)$, where $\tilde{\boldsymbol{Y}}_i$ and $\tilde{\boldsymbol{X}}^0$ are the inputs of the K-layer RNN. Given $\{\tilde{\boldsymbol{X}}_i^*, \tilde{\boldsymbol{Y}}_i\}_{i=1}^T$, we obtain the parameters of K-layer RNN via solving the following problem

$$\boldsymbol{\Theta}^* = \arg\min_{\boldsymbol{\Theta}} \sum_{i=1}^T \left\|\tilde{\boldsymbol{X}}^K(\boldsymbol{\Theta}, \tilde{\boldsymbol{Y}}_i, \tilde{\boldsymbol{X}}^0) - \tilde{\boldsymbol{X}}_i^*\right\|_F^2. \tag{36}$$

To avoid converging to a local minimum, the network parameters are trained layer-by-layer [22]. We take the training of the parameters of layer k, denoted as $\mathbf{\Theta}_{k-1}$, as an example, which is performed after the parameters of the first $(k-1)$ layers, denoted as $\mathbf{\Theta}_{0:k-2}$, are trained. To optimize $\mathbf{\Theta}_{k-1}$, we need to solve problem

$$\min_{\mathbf{\Theta}_{k-1}} \sum_{i=1}^{T} \|\tilde{\boldsymbol{X}}^{k}(\mathbf{\Theta}_{0:k-1}, \tilde{\boldsymbol{Y}}_i, \tilde{\boldsymbol{X}}^{0}) - \tilde{\boldsymbol{X}}_i^{*}\|_F^2 \tag{37}$$

with learning rate α_0. After that, we further solve problem

$$\min_{\mathbf{\Theta}_{0:k-1}} \sum_{i=1}^{T} \|\tilde{\boldsymbol{X}}^{k}(\mathbf{\Theta}_{0:k-1}, \tilde{\boldsymbol{Y}}_i, \tilde{\boldsymbol{X}}^{0}) - \tilde{\boldsymbol{X}}_i^{*}\|_F^2 \tag{38}$$

to optimize parameters $\mathbf{\Theta}_{0:k-1}$ with learning rates α_1 and α_2. Through the above process, the first k layers' parameters can be obtained. After learning these parameters, the BS performs JADCE in the test stage by applying the proposed unfolding networks.

2) LPGM-AT: For LPGM-AT, we only need to find the appropriate hyperparameters (i.e. c_θ, c_β, and c_η) by using grid search in the training stage, which significantly reduces the training cost.

IV. Simulation Results

In the simulations, the channels between the BS and IoT devices follow independent Rayleigh fading. The activity of each device follows an independent Bernoulli distribution. We set $\mathbb{P}(a_n = 0) = 0.9$ and $\mathbb{P}(a_n = 1) = 0.1$, $\forall\, n \in [N]$. We set the regularization parameter λ as 0.1 and define the transmit SNR as $\mathbb{E}[\|\boldsymbol{S}\boldsymbol{X}\|_F^2]/\mathbb{E}[\|\boldsymbol{Z}\|_F^2]$. The neural networks have $K = 16$ layers. The sizes of training dataset, validation dataset, and test dataset are 51200, 2048, and 2048, respectively. The learning rates are set to $\alpha_0 = 1\times10^{-3}$, $\alpha_1 = 0.2\alpha_0$, and $\alpha_2 = 0.02\alpha_0$. In the test phase, the group-sparse-matrix recovery performance is measured by using the normalized mean square error (NMSE), defined as $\mathrm{NMSE}(\tilde{\boldsymbol{X}}^{k}, \tilde{\boldsymbol{X}}^{*}) = 10\log_{10}\left(\mathbb{E}\|\tilde{\boldsymbol{X}}^{k} - \tilde{\boldsymbol{X}}^{*}\|_F^2/\mathbb{E}\|\tilde{\boldsymbol{X}}^{*}\|_F^2\right)$.

A. Performance Comparison

In the first part of the simulation, ALPGM is compared with the following two unfolding PGM:

- Step-LPGM: By fixing $\tilde{\boldsymbol{S}}^{\mathrm{T}}$ in (9) and denoting $\theta_k = \lambda\gamma_k$, we learn the step-size γ_k, thresholding parameter θ_k, and parameter η_k. The trainable parameters are the same as that of ALPGM. The neural network is given by
$$\tilde{\boldsymbol{X}}^{k+1} = P_{\theta_k, f_{\eta_k}}\left(\tilde{\boldsymbol{X}}^{k} + \gamma_k\tilde{\boldsymbol{S}}^{\mathrm{T}}(\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^{k})\right), \quad k = 0, \ldots, K-1.$$
- LPGM-CP: We replace $\gamma_k\tilde{\boldsymbol{S}}^{\mathrm{T}}$ in (9) by $\boldsymbol{B}^k$ and obtain the following neural network
$$\tilde{\boldsymbol{X}}^{k+1} = P_{\theta_k, f_{\eta_k}}\left(\tilde{\boldsymbol{X}}^{k} + \boldsymbol{B}^{k}(\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^{k})\right), \quad k = 0, \ldots, K-1,$$
where $\{\boldsymbol{B}^k, \theta_k, \eta_k\}$ are trainable parameters.

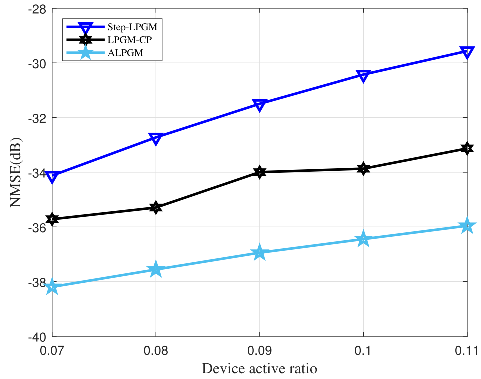


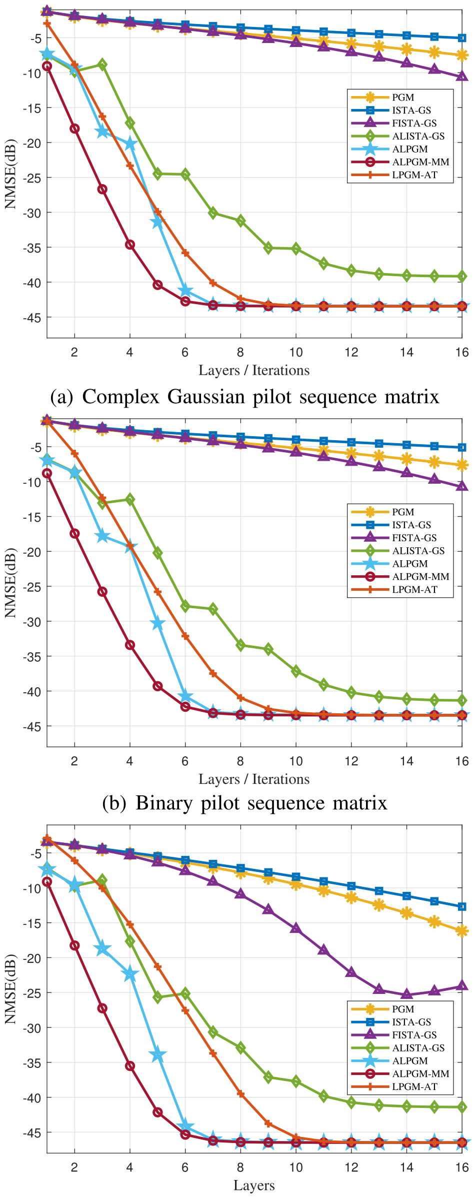
Fig. 6. NMSE versus device active ratio for different proximal gradient methods when $L = 45$, $N = 90$, $M = 4$, and SNR $= 30$ dB.

We utilize Zadoff-Chu pilot sequence matrix [45] and generate it as in [46]. Each column of the pilot sequence matrix is normalized. Fig. 6 shows that our proposed ALPGM achieves a smaller NMSE than other methods. This is because, in ALPGM, $\tilde{\boldsymbol{S}}$ is replaced by matrix $\boldsymbol{B}^{\mathrm{T}}$ that has a small 'generalized coherence' with $\tilde{\boldsymbol{S}}$, resulting in a performance gain.

B. Convergence Performance

Unfolding PGM has shown its better performance than ISTA and LISTA for solving SMV problems in [29]. Thus, we in this paper do not compare the unfolding PGM with these methods for regular sparse recovery. We focus on comparing our proposed structures with the following methods for group sparsity:

- **PGM**: PGM is an iterative algorithm to solve MMV problems. The update formula of PGM is given in (9).
- **ISTA-GS**: ISTA-GS [8] is an extension of ISTA to solve MMV problems by replacing the scalar soft-thresholding function in ISTA with multidimensional shrinkage thresholding operator. The update formula of ISTA-GS is given by
$$\tilde{\boldsymbol{X}}^{k+1} = \mathcal{T}_{\lambda/C}\left(\tilde{\boldsymbol{X}}^{k} + \frac{1}{C}\tilde{\boldsymbol{S}}^{\mathrm{T}}(\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^{k})\right), \tag{39}$$
where $\mathcal{T}_{\lambda/C}$ is the multidimensional shrinkage thresholding operator
$$\mathcal{T}_{\theta}(\tilde{\boldsymbol{X}}_{i,:}) = \max\{0, \|\tilde{\boldsymbol{X}}_{i,:}\|_2 - \theta\}\frac{\tilde{\boldsymbol{X}}_{i,:}}{\|\tilde{\boldsymbol{X}}_{i,:}\|_2} \tag{40}$$
with $\theta = \lambda/C$, $\lambda = 0.1$, and C denotes the largest eigenvalue of $\tilde{\boldsymbol{S}}^{\mathrm{T}}\tilde{\boldsymbol{S}}$.
- **Fast ISTA-GS (FISTA-GS)**: FISTA [47] is a Nesterov momentum speed-up of ISTA. Correspondingly, FISTA-GS is an accelerated variant of ISTA-GS to solve MMV problems.
- **ALISTA-GS**: ALISTA-GS is an unfolding algorithm for MMV problems proposed in [25]. The neural network is
$$\tilde{\boldsymbol{X}}^{k+1} = \mathcal{T}_{\theta_k}\left(\tilde{\boldsymbol{X}}^{k} + \gamma_k\boldsymbol{B}(\tilde{\boldsymbol{Y}} - \tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^{k})\right), \tag{41}$$
where matrix $\boldsymbol{B}$ can be obtained by solving problem (13), and $\{\theta_k, \gamma_k\}$ are trainable parameters. Other settings

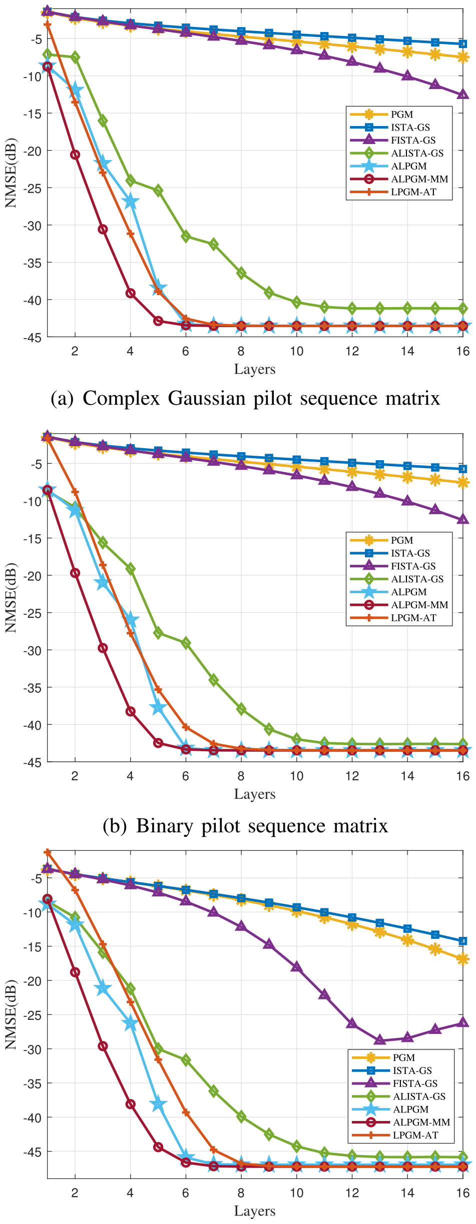


(a) Complex Gaussian pilot sequence matrix

(b) Binary pilot sequence matrix

(c) Zadoff-Chu pilot sequence matrix

Fig. 7. NMSE versus number of layers or iterations for different pilot sequence matrices when $M = 6$, $N = 250$, and $L = 125$.



(a) Complex Gaussian pilot sequence matrix

(b) Binary pilot sequence matrix

(c) Zadoff-Chu pilot sequence matrix

Fig. 8. NMSE versus number of layers or iterations for different pilot sequence matrices when $M = 10$, $N = 300$, and $L = 175$.

of ALISTA-GS are the same as that of our proposed algorithms.

- **SPARROW**: The authors in [48] formulated a $\ell_{2,1}$ minimization problem as a semidefinite program (SDP) problem. According to Section VII in [48], we utilize MOSEK solver with CVX MATLAB toolbox to solve the SDP problem. We run SPARROW for 300 times and take the average as its performance.

We evaluate these methods using three types of pilot sequence matrices, i.e., complex Gaussian pilot sequence matrix, binary pilot sequence matrix, and Zadoff-Chu pilot sequence matrix. Specifically, we generate the complex Guassian pilot sequence matrix by utilizing the complex Gaussian distribution. For the binary pilot sequence matrix, each element is selected uniformly at random on 1 or -1. In addition, each column of the pilot sequence matrix is normalized. The SNR is set to 40 dB.

Fig. 7 and Fig. 8 depict the NMSE versus number of layers or iterations for our proposed networks and the baseline methods under different settings of M, N, and L. ALPGM achieves much lower NMSE than PGM because the parameters in ALPGM are learned to fit the target signals. Benefiting from the MCP-based multivariate proximal operator, the proposed networks (i.e., ALPGM, ALPGM-MM, and LPGM-AT) achieve better performance and faster convergence rate than the baseline methods under all three pilot sequence matrices. Besides, ALPGM-MM achieves faster convergence rate than ALPGM because the momentum term accelerates convergence. In this experiment, the test data has the same distribution as the training data, and thus the proposed

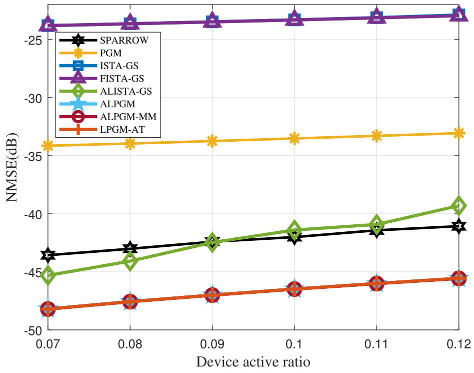


Fig. 9. NMSE versus device active ratio.

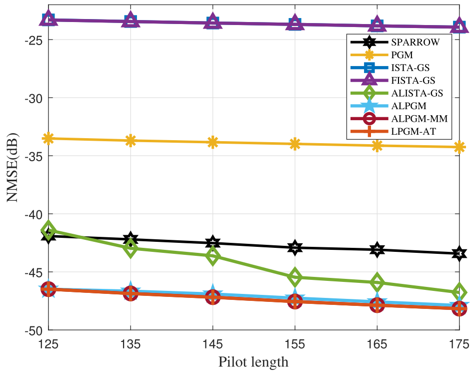


Fig. 10. NMSE versus pilot length.

APGM-AT achieves almost the same performance as ALPGM and ALPGM-MM.

C. Performance Comparison Under Different Settings

We compare our proposed three networks with the baselines under various device active ratios, lengths of pilot, and SNRs. As the Zadoff-Chu pilot sequence matrix outperforms other practical pilot sequence matrices in Fig. 7 and Fig. 8, in the following we adopt the Zadoff-Chu pilot sequence matrix. In order to ensure the convergence of iterative methods (i.e., PGM, ISTA-GS, and FISTA-GS), the numbers of iterations of iterative methods are set to 50. The number of layers of all neural networks is 16. Other settings are same as Section IV-B.

In Figs. 9, 10, and 11, we observe that ISTA-GS and FISTA-GS achieve similar performance after convergence. In addition, PGM outperforms ISTA-GS because the MCP-based proximal operator is more capable of inducing sparsity than ℓ_1-norm. The proposed three networks all achieve much lower NMSEs than the baseline methods under different device active ratios, lengths of pilot, and SNRs. In Fig. 10, the NMSE decreases with the pilot length, as a longer pilot sequence generally leads to a smaller level of non-orthogonality, making it easier to recover the orginal sparse signal. The results in Fig. 11 demonstrate that by utilizing MCP, the proposed ALPGM-MM reduces the NMSE 12% compared to ALISTA-GS when SNR = 50 dB. Besides, when the test dataset shares the same distribution with the training dataset, the proposed LPGM-AT achieves comparable performance with ALPGM and ALPGM-MM. Fig. 12 shows NMSE versus the number of devices with $L = 45$, $M = 6$, and SNR = 40 dB. We observe that the NMSE increases as the number of devices increases for all the methods. Besides, the proposed networks obtain lower NMSE compared to the baseline methods.

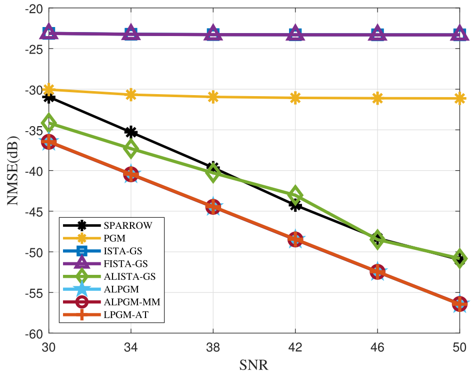


Fig. 11. NMSE versus SNR.

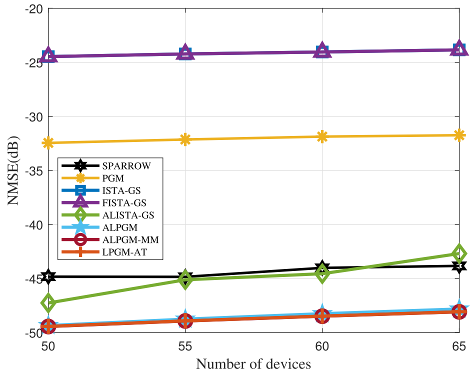


Fig. 12. NMSE versus number of devices.

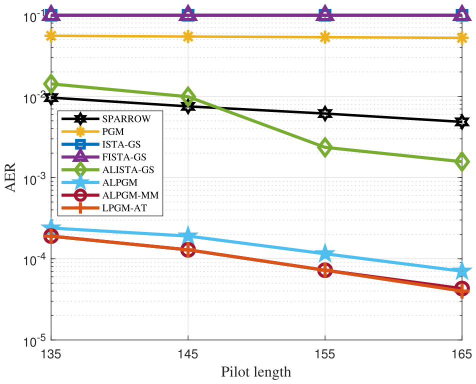


Fig. 13. AER versus pilot length.

We evaluate the device activity detection performance by using the activity error ratio (AER), which is defined as $\mathrm{AER}(\tilde{\boldsymbol{X}}^k, \tilde{\boldsymbol{X}}^*) = \mathbb{E}\|\psi(\tilde{\boldsymbol{X}}^k) - \psi(\tilde{\boldsymbol{X}}^*)\|_0/2N$. Fig. 13 shows that the AER of our proposed unfolding neural networks is smaller than 10^{-3}, which is smaller than that of baseline methods. The results imply that using non-convex penalty MCP can significantly improve the performance of detecting active devices.

TABLE I
SPECTRAL EFFICIENCY COMPARISON

	ALPGM	ALPGM-MM	LPGM-AT	ALISTA-GS	PGM	ISTA-GS	FISTA-GS
Sum-Rate (bps/Hz)	18.3374	18.2764	17.9504	16.3857	11.7264	11.5446	11.6470

TABLE II
TRAINING TIME AND TEST TIME COMPARISON

	ALPGM	ALPGM-MM	LPGM-AT	ALISTA-GS	PGM	ISTA-GS	FISTA-GS	SPARROW
Training Time	2.49 h	2.25 h	8.18 min	3.27 h	-	-	-	-
Test Time per Sample	6.2×10^{-4} s	6.5×10^{-4} s	6.4×10^{-4} s	6.0×10^{-4} s	2.5×10^{-3} s	2.3×10^{-3} s	2.5×10^{-3} s	58.63 s

D. Spectral Efficiency Comparison

We consider coherence time of length T consisting of two phases: JADCE of length L, and uplink data transmission of length $T - L$. Specifically, the active IoT devices send pilot sequences to the BS during the first L symbols in the JADCE phase. After JADCE, the active IoT devices send data messages to the BS during the remaining $T - L$ symbols. We denote the set of active devices within a coherence block as $\mathcal{K}$. The received signal at the BS in the uplink data transmission phase is given by

$$\boldsymbol{y}^u = \sum_{n \in \mathcal{K}} \boldsymbol{h}_n \sqrt{p_n^u} s_n^u + \boldsymbol{z}^u, \tag{42}$$

where s_n^u denotes the transmitted symbol of device $n \in \mathcal{K}$ following Gaussian distribution with unit norm, p_n^u denotes the uplink transmit power, and $\boldsymbol{z}^u$ denotes the AWGN vector at the BS with each element following distribution $\mathcal{CN}(0, \sigma^2)$.

The BS utilizes minimum mean squared error (MMSE) beamforming based on the estimated channel $\hat{\boldsymbol{h}}_n$ on the received signal to decode the transmitted symbol. The MMSE beamformer [34] for device n is given by

$$\boldsymbol{v}_n = \left(\sum_{m \in \mathcal{K}} p_m^u \hat{\boldsymbol{h}}_m \hat{\boldsymbol{h}}_m^{\mathrm{H}} + \sigma^2 \boldsymbol{I} \right)^{-1} \hat{\boldsymbol{h}}_n. \tag{43}$$

Then, the decoded symbol is given by

$$\hat{s}_n^u = \boldsymbol{v}_n^{\mathrm{H}} \boldsymbol{h}_n \sqrt{p_n^u} s_n^u + \sum_{m \in \mathcal{K}/k} \boldsymbol{v}_n^{\mathrm{H}} \boldsymbol{h}_m \sqrt{p_m^u} s_m^u + \boldsymbol{v}_n^{\mathrm{H}} \boldsymbol{z}^u. \tag{44}$$

Hence, the achievable rate is given by

$$r_n^u = \frac{T - L}{T} \log_2(1 + \gamma_n^u), \tag{45}$$

where γ_n^u denotes the signal-to-interference-plus-noise ratio (SINR) defined as

$$\gamma_n^u = \frac{p_n^u |\boldsymbol{v}_n^{\mathrm{H}} \boldsymbol{h}_n|^2}{\sum_{m \in \mathcal{K}/k} p_m^u |\boldsymbol{v}_n^{\mathrm{H}} \boldsymbol{h}_m|^2 + \sigma^2}. \tag{46}$$

We set $L = 35$, $N = 70$, $M = 6$ and $T = 200$. The uplink transmission power is set to 20 dBm and the number of layers or iterations is set to 5. In Table I, we observe that our proposed networks achieve better spectral efficiency than the baseline methods due to lower channel estimation error during JADCE phase.

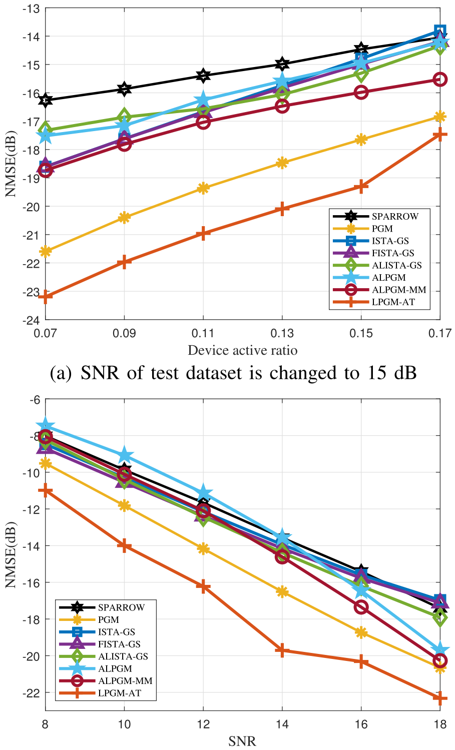
E. Adaptation Comparison

We compare the adaptivity of the proposed three networks the baseline methods for the scenario with mismatch between the training and test datasets. In this subsection, the number of iterations of iterative methods (i.e., PGM, ISTA-GS, and FISTA-GS) is set to 50, while the number of the layers of the DL-based methods (i.e., ALPGM, ALPGM-MM, and ALISTA-GS) and LPGM-AT is set to 16. ALPGM, ALPGM-MM, and ALISTA-GS are trained by back-propagation under the settings of Fig. 7(c). LPGM-AT is trained by grid search to find the best hyperparameter combination on the same training dataset as ALPGM, ALPGM-MM, and ALISTA-GS. Then we directly apply them to the test dataset with different device active ratios and SNRs.

Fig. 14(a) shows the NMSE versus the device active ratio of the test dataset when SNR = 15 dB for the test dataset. The performance of the DL-based methods (i.e., ALPGM, ALPGM-MM, and ALISTA-GS) degrades because they are sensitive to the mismatch between the training and test datasets. However, the results clearly show that LPGM-AT outperforms the DL-based methods and iterative methods because LPGM-AT can adapt its parameters to different distributions of the test dataset. In Fig. 14(b), we change the device active ratio of the test dataset to 0.15 with lower SNRs than that of the training dataset. The results indicate that LPGM-AT is able to adapt to time-varying IoT networks and outperforms other methods.

F. Computation Complexity Comparison

In this subsection, the training and test time of the proposed three networks are compared with the baseline methods. Table II shows that the training time of the DL-based methods (i.e., ALPGM, ALPGM-MM, and ALISTA-GS) need several hours. This is because DL-based methods optimize parameters by back-propagation on a large volume of training data, and the training procedures are time-consuming. Since the momentum term provides convergence acceleration, ALPGM-MM has the least training time among the DL-based methods. In contrast, the grid search for LPGM-AT is quite computation-efficient, because it only need to search three hyperparameters. LPGM-AT only needs less than 10 minutes to find the best hyperparameters by grid search, which dramatically reduces the computation overhead. Moreover, the test



(a) SNR of test dataset is changed to 15 dB

(b) Device active ratio of test dataset is changed to $\mathbb{P}(a_n = 1) = 0.15$

Fig. 14. All models except iterative methods are trained when $\boldsymbol{S}$ is Zadoff-Chu pilot sequence matrix and the device active ratio is 0.1 with SNR = 40 dB.

time of the proposed network is much smaller than the iterative methods, which demonstrates that the proposed network is more practical for JADCE in IoT networks.

V. Conclusion

In this paper, we proposed an unfolding framework that is based on PGM for massive random access. We first mapped PGM as an unfolding neural network to reduce the computational complexity. In order to further improve the convergence rate, we embedded momentum into the unfolding neural network, and proved accelerated convergence theoretically. Based on the convergence analysis, we developed an adaptive network that generalizes well to different device active ratios and SNRs by adjusting its network parameters. Simulation results showed that the proposed unfolding framework achieves greater recovery performance, faster convergence, and better adaptivity than the baselines. For further studies, we will extend this work to investigate the impact of the antenna correlation on the JADCE performance and evaluate the impact of the proposed unfolding algorithms on the spectral efficiency of the subsequent data transmission.

Appendix

A. Proof of Theorem 1

We assume that the noise level ϵ satisfies

$$\epsilon \le \min\left(\frac{\mu_x}{3\mu_B}, \frac{\mu_x(1-c_{\phi s})}{6\mu_B(1+s)}, \frac{\sqrt{s}\mu_x(1-c_{\phi s})(c_{\phi s})^{K_0}}{(1+s+\sqrt{s})\mu_B}\right), \tag{47}$$

where $\hat{\beta}$, C_0, and K_0 are defined as

$$\hat{\beta} \triangleq \frac{1}{2s}\left(1-\sqrt{1-c_{\phi s}}\right)^2, \tag{48}$$

$$C_0 \triangleq \max\left(s\mu_x, \frac{8\mu_x s\sqrt{s}(1+\hat{\beta})}{c_{\phi s}\sqrt{4\hat{\beta}-(\hat{\beta}+\phi s-\phi)^2}}\right), \tag{49}$$

$$K_0 \triangleq \left\lceil \frac{\log(\mu_x)-\log(6C_0)}{\log(c_{\phi s})} \right\rceil + 1. \tag{50}$$

Then, we set the specific conditions of parameters $\{\beta_k, \eta_k\}$ as follows

$$\beta_k = \begin{cases} 0, & \text{if}\, k \le K_0, \\ \hat{\beta}, & \text{if}\, k \ge K_0+1, \end{cases} \tag{51}$$

$$\eta_k \begin{cases} < \dfrac{1}{2\theta_k}, & \text{if}\, k \le K_0-1, \\ = \dfrac{1}{2\theta_k}, & \text{if}\, k \ge K_0. \end{cases} \tag{52}$$

1) Proof of No-False-Positive Property: When $k=0$ and $\tilde{\boldsymbol{X}}^0 = \boldsymbol{0}$, we have $\text{supp}(\psi(\tilde{\boldsymbol{X}}^0)) = \emptyset \subseteq S$. We fix $k \ge 0$ and assume $\boldsymbol{X}^t_{i,:} = 0, \forall i \notin S, 0 \le t \le k$. For $\forall i \notin S$, we have

$$\begin{aligned}
&\|-\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:}-\tilde{\boldsymbol{X}}^*_{S,:})\|_2 \\
&= \|-\sum_{l\in S}\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,l}(\tilde{\boldsymbol{X}}^k_{l,:}-\tilde{\boldsymbol{X}}^*_{l,:})\|_2 \\
&\le \sum_{l\in S}\|\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,l}(\tilde{\boldsymbol{X}}^k_{l,:}-\tilde{\boldsymbol{X}}^*_{l,:})\|_2 \le \sum_{l\in S}|\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,l}|\|\tilde{\boldsymbol{X}}^k_{l,:}-\tilde{\boldsymbol{X}}^*_{l,:}\|_2 \\
&\overset{(a)}{\le} \sum_{l\in S}\phi\|\tilde{\boldsymbol{X}}^k_{l,:}-\tilde{\boldsymbol{X}}^*_{l,:}\|_2 \le \phi\|\tilde{\boldsymbol{X}}^k-\tilde{\boldsymbol{X}}^*\|_{2,1},
\end{aligned} \tag{53}$$

where (a) is due to $\phi \ge |(\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D})_{i,l}| = |\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,l}|$. Since $\|\boldsymbol{B}\|_F \le \|\boldsymbol{B}\|_{2,1} \le \mu_B$ and $\|\tilde{\boldsymbol{Z}}\|_F \le \epsilon$, we have

$$\|\boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}}\|_2 \le \|\boldsymbol{B}_{i,:}\|_2\|\tilde{\boldsymbol{Z}}\|_F \le \|\boldsymbol{B}\|_F\|\tilde{\boldsymbol{Z}}\|_F \le \mu_B\epsilon. \tag{54}$$

By combining (53) and (54), we obtain the lower bound for the threshold parameter θ_k

$$\begin{aligned}
\frac{1}{2\eta_k} &> \theta_k \ge \phi\|\tilde{\boldsymbol{X}}^k-\tilde{\boldsymbol{X}}^*\|_{2,1}+\mu_B\epsilon \\
&\ge \|-\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:}-\tilde{\boldsymbol{X}}^*_{S,:})\|_2+\|\boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}}\|_2 \\
&\ge \|-\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:}-\tilde{\boldsymbol{X}}^*_{S,:})+\boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}}\|_2.
\end{aligned} \tag{55}$$

From the update formula of ALPGM-MM, for $\forall i \notin S$, we have

$$\begin{aligned}
\tilde{\boldsymbol{X}}^{k+1}_{i,:} &= P_{\theta_k,f_{\eta_k}}\left(\tilde{\boldsymbol{X}}^k_{i,:}-\boldsymbol{B}_{i,:}(\tilde{\boldsymbol{S}}\tilde{\boldsymbol{X}}^k-\tilde{\boldsymbol{Y}})+\beta^k(\tilde{\boldsymbol{X}}^k_{i,:}-\tilde{\boldsymbol{X}}^{k-1}_{i,:})\right) \\
&= P_{\theta_k,f_{\eta_k}}\Big(\tilde{\boldsymbol{X}}^k_{i,:}-\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:}-\tilde{\boldsymbol{X}}^*_{S,:}) \\
&\quad +\boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}}+\beta_k(\tilde{\boldsymbol{X}}^k_{i,:}-\tilde{\boldsymbol{X}}^{k-1}_{i,:})\Big).
\end{aligned} \tag{56}$$

As $\boldsymbol{X}^t_{i,:} = 0, \forall i \notin S, 0 \le t \le k$, we obtain

$$\begin{aligned}
\tilde{\boldsymbol{X}}^{k+1}_{i,:} &= P_{\theta_k,f_{\eta_k}}\left(-\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:}-\tilde{\boldsymbol{X}}^*_{S,:})+\boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}}\right) \\
&= \hat{P}_{\theta_k,f_{\eta_k}}(\|\boldsymbol{v}\|_2)\frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_2},
\end{aligned} \tag{57}$$

where $\boldsymbol{v} = -\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:} - \boldsymbol{X}^*_{S,:}) + \boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}}$. According to (55) and (11), we obtain $\tilde{\boldsymbol{X}}^{k+1}_{i,:} = 0, \forall\, i \notin S$. By induction, we complete the proof.

2) Convergence Analysis: Firstly, we analyze the convergence when $\beta_k = 0$. When $\beta_k = 0$ and $k \le K_0$, ALPGM-MM reduces to ALPGM. By the definition of multivariate proximal operator, for $\forall\, i \in S$, we have

$$\begin{aligned}\tilde{\boldsymbol{X}}^{k+1}_{i,:} &= P_{\theta_k, f_{\eta_k}}\Big(\tilde{\boldsymbol{X}}^k_{i,:} - \boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:} - \tilde{\boldsymbol{X}}^*_{S,:}) \\ &\quad + \boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}} + \beta_k(\tilde{\boldsymbol{X}}^k_{i,:} - \tilde{\boldsymbol{X}}^{k-1}_{i,:})\Big) \\ &= \arg\min_{\tilde{\boldsymbol{U}}_{i,:}} \frac{1}{2}\|\tilde{\boldsymbol{U}}_{i,:} - (\tilde{\boldsymbol{X}}^k_{i,:} - \boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:} - \tilde{\boldsymbol{X}}^*_{S,:}) \\ &\quad + \boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}} + \beta_k(\tilde{\boldsymbol{X}}^k_{i,:} - \tilde{\boldsymbol{X}}^{k-1}_{i,:}))\|_2^2 + \theta_k g_{\eta_k}(\|\tilde{\boldsymbol{U}}_{i,:}\|_2).\end{aligned} \tag{58}$$

According to the optimality condition, we have

$$\begin{aligned}\boldsymbol{0} \in \tilde{\boldsymbol{X}}^{k+1}_{i,:} &- \Big(\tilde{\boldsymbol{X}}^k_{i,:} - \boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:} - \tilde{\boldsymbol{X}}^*_{S,:}) \\ &+ \boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}} + \beta_k(\tilde{\boldsymbol{X}}^k_{i,:} - \tilde{\boldsymbol{X}}^{k-1}_{i,:})\Big) + \theta_k \partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2),\end{aligned} \tag{59}$$

where $\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2)$ is the subgradient of $g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2)$.

Recalling the definition of $g_\eta(\cdot)$, we have

$$\begin{aligned}&g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2) \\ &= \Big(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 - \eta_k\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2^2\Big)\mathbb{1}_{\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 \le \frac{1}{2\eta_k}}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2) \\ &\quad + \Big(\frac{1}{4\eta_k}\Big)\mathbb{1}_{\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 > \frac{1}{2\eta_k}}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2).\end{aligned} \tag{60}$$

One can easily check that

$$\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2) = \Big(\partial\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 - 2\eta_k\tilde{\boldsymbol{X}}^{k+1}_{i,:}\Big)\mathbb{1}_{\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 \le \frac{1}{2\eta_k}}, \tag{61}$$

where

$$\partial\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 = \begin{cases}\dfrac{\tilde{\boldsymbol{X}}^{k+1}_{i,:}}{\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2}, & \text{if } \tilde{\boldsymbol{X}}^{k+1}_{i,:} \ne \boldsymbol{0}, \\ \{\boldsymbol{h} \in \mathbb{R}^{1\times M} | \|\boldsymbol{h}\|_2 \le 1\}, & \text{otherwise.}\end{cases} \tag{62}$$

Hence, we obtain

$$\begin{aligned}&\|\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2)\|_2^2 \\ &= \Big(\|\boldsymbol{h}\|_2^2\Big)\mathbb{1}_{\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 = 0}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2) \\ &\quad + \left(\left\|\frac{\tilde{\boldsymbol{X}}^{k+1}_{i,:}}{\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2} - 2\eta_k\tilde{\boldsymbol{X}}^{k+1}_{i,:}\right\|_2^2\right)\mathbb{1}_{0 < \|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 \le \frac{1}{2\eta_k}}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2) \\ &\le \mathbb{1}_{\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 = 0}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2) + \mathbb{1}_{0 < \|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2 \le \frac{1}{2\eta_k}}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2) = 1.\end{aligned} \tag{63}$$

From (59), we have

$$\begin{aligned}&\tilde{\boldsymbol{X}}^{k+1}_{i,:} - \tilde{\boldsymbol{X}}^*_{i,:} \\ &= \tilde{\boldsymbol{X}}^k_{i,:} - \tilde{\boldsymbol{X}}^*_{i,:} - \boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^k_{S,:} - \tilde{\boldsymbol{X}}^*_{S,:}) + \boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}} \\ &\quad - \theta_k\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2) = -\sum_{j\in S,\, j\ne i}\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,j}(\tilde{\boldsymbol{X}}^k_{j,:} - \tilde{\boldsymbol{X}}^*_{j,:}) \\ &\quad + \boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}} - \theta_k\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2),\end{aligned} \tag{64}$$

where the last equality follows from the constraint $(\boldsymbol{D}^{\mathrm{T}}\boldsymbol{D})_{i,i} = \boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,i} = 1$.

We take norm on both sides of (64) and obtain

$$\begin{aligned}&\|\tilde{\boldsymbol{X}}^{k+1}_{i,:} - \tilde{\boldsymbol{X}}^*_{i,:}\|_2 \\ &\le \sum_{j\in S,\, j\ne i}|\boldsymbol{B}_{i,:}\tilde{\boldsymbol{S}}_{:,j}|\|\tilde{\boldsymbol{X}}^k_{j,:} - \tilde{\boldsymbol{X}}^*_{j,:}\|_2 + \|\boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}}\|_2 \\ &\quad + \theta_k\|\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{i,:}\|_2)\|_2 \\ &\le \phi\sum_{j\in S,\, j\ne i}\|\tilde{\boldsymbol{X}}^k_{j,:} - \tilde{\boldsymbol{X}}^*_{j,:}\|_2 + \|\boldsymbol{B}_{i,:}\tilde{\boldsymbol{Z}}\|_2 + \theta_k.\end{aligned} \tag{65}$$

Due to the no-false-positive property, we obtain $\|\tilde{\boldsymbol{X}}^{k+1} - \tilde{\boldsymbol{X}}^*\|_{2,1} = \|\tilde{\boldsymbol{X}}^{k+1}_{S,:} - \tilde{\boldsymbol{X}}^*_{S,:}\|_{2,1}$ and

$$\|\tilde{\boldsymbol{X}}^{k+1} - \tilde{\boldsymbol{X}}^*\|_{2,1} \le \phi(|S|-1)\|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_{2,1} + \mu_B\epsilon + |S|\theta_k. \tag{66}$$

By taking supremum on both sides of inequality (66), we obtain

$$\begin{aligned}&\sup_{\substack{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \\ \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)}}\|\tilde{\boldsymbol{X}}^{k+1} - \tilde{\boldsymbol{X}}^*\|_{2,1} \\ &\le \phi(s-1)\sup_{\substack{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \\ \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)}}\|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_{2,1} + \mu_B\epsilon + s\theta_k.\end{aligned} \tag{67}$$

By plugging the definition of θ_k into (67), we obtain

$$\begin{aligned}&\sup_{\substack{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \\ \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)}}\|\tilde{\boldsymbol{X}}^{k+1} - \tilde{\boldsymbol{X}}^*\|_{2,1} \\ &\le \phi(s-1)\sup_{\substack{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \\ \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)}}\|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_{2,1} + \mu_B\epsilon \\ &\quad + \phi s\sup_{\substack{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \\ \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)}}\|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_{2,1} + s\mu_B\epsilon \\ &= (2\phi s - \phi)\sup_{\substack{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \\ \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)}}\|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_{2,1} + (1+s)\mu_B\epsilon.\end{aligned} \tag{68}$$

We denote $e^k = \sup_{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)}\|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_{2,1}$. If $(2s-1)\phi < 1$, we obtain

$$\begin{aligned}e^{k+1} &\le (c_{\phi s})e^k + (1+s)\mu_B\epsilon \\ &\le (c_{\phi s})^{k+1}e^0 + \Big(\sum_{t=0}^{k}(c_{\phi s})^t\Big)(1+s)\mu_B\epsilon\end{aligned}$$

$$\le (c_{\phi s})^{k+1} e^0 + \frac{(1+s)\mu_B \epsilon}{1-c_{\phi s}}. \tag{69}$$

As $\tilde{\boldsymbol{X}}^0 = 0$, we obtain

$$e^0 = \sup_{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)} \|\tilde{\boldsymbol{X}}^*\|_{2,1} \le s\mu_x \le C_0. \tag{70}$$

Since $\|\boldsymbol{X}\|_F \le \|\boldsymbol{X}\|_{2,1}$, we conclude with

$$\|\tilde{\boldsymbol{X}}^k - \tilde{\boldsymbol{X}}^*\|_F \le C_0 (c_{\phi s})^k + \frac{(1+s)\mu_B\epsilon}{1-c_{\phi s}}, \quad \forall\, k \le K_0 + 1. \tag{71}$$

Secondly, we analyze the convergence when $\beta_k = \hat{\beta}$. We define $\bar{\boldsymbol{X}}_{S,:} = \tilde{\boldsymbol{X}}^*_{S,:} + \delta\tilde{\boldsymbol{X}}_{S,:}$, where $\delta\tilde{\boldsymbol{X}}_{S,:} = (\boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S})^{-1}\boldsymbol{B}_{S,:}\tilde{\boldsymbol{Z}}$. Then, we obtain

$$\begin{aligned} &\|(\boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S})^{-1}\|_F \overset{(a)}{\le} \sqrt{|S|}\|(\boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S})^{-1}\|_2 \\ &= \frac{\sqrt{|S|}}{\sigma_{\min}(\boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S})} \overset{(b)}{\le} \frac{\sqrt{|S|}}{1+\phi-\phi|S|} \le \frac{\sqrt{s}}{1+\phi-\phi s}, \end{aligned} \tag{72}$$

where (a) is due to $\|\boldsymbol{A}\|_F^2 = \sum_{i=1}^n \sigma_i^2(\boldsymbol{A}) \le n\sigma^2_{\max}(\boldsymbol{A}) = n\|\boldsymbol{A}\|_2^2$ with $\sigma_i(\boldsymbol{A})$ denoting the singular value of the matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, and (b) follows by Gershgorin circle theorem [49, Chapter 7]. Hence, we obtain

$$\|\delta\tilde{\boldsymbol{X}}_{S,:}\|_F \le \|(\boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S})^{-1}\|_F \|\boldsymbol{B}_{S,:}\tilde{\boldsymbol{Z}}\|_F \le \frac{\sqrt{s}\mu_B\epsilon}{1+\phi-\phi s}. \tag{73}$$

From (59), for $k \ge \tilde{K_0} + 1$, we have

$$\begin{aligned} \tilde{\boldsymbol{X}}^{k+1}_{S,:} = \tilde{\boldsymbol{X}}^{k}_{S,:} - \boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}^{k}_{S,:} - \bar{\boldsymbol{X}}_{S,:}) + \beta_k(\tilde{\boldsymbol{X}}^{k}_{S,:} - \tilde{\boldsymbol{X}}^{k-1}_{S,:}) \\ - \theta_k \partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{S,:}\|_2), \end{aligned} \tag{74}$$

where $\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{2N,:}\|_2)$ is defined as

$$\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{2N,:}\|_2) = \left[\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{1,:}\|_2)^{\mathrm{T}}, \ldots, \partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{2N,:}\|_2)^{\mathrm{T}}\right]^{\mathrm{T}}. \tag{75}$$

By substracting $\bar{\boldsymbol{X}}_{S,:}$ from both sides of (74), we obtain

$$\begin{aligned} &\tilde{\boldsymbol{X}}^{k+1}_{S,:} - \bar{\boldsymbol{X}}_{S,:} \\ &= \Big((1+\beta_k)\boldsymbol{I}_S - \boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S}\Big)(\tilde{\boldsymbol{X}}^{k}_{S,:} - \bar{\boldsymbol{X}}_{S,:}) \\ &\quad -\beta_k(\tilde{\boldsymbol{X}}^{k-1}_{S,:} - \bar{\boldsymbol{X}}_{S,:}) - \theta_k\partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{S,:}\|_2). \end{aligned} \tag{76}$$

By plugging $\beta_k = \hat{\beta}$ with $k \ge K_0 + 1$ into (76), we obtain

$$\begin{aligned} \underbrace{\begin{bmatrix} \tilde{\boldsymbol{X}}^{k+1}_{S,:} - \bar{\boldsymbol{X}}_{S,:} \\ \tilde{\boldsymbol{X}}^{k}_{S,:} - \bar{\boldsymbol{X}}_{S,:} \end{bmatrix}}_{\boldsymbol{z}^k} &= \underbrace{\begin{bmatrix} (1+\hat{\beta})\boldsymbol{I}_S - \boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S} & -\hat{\beta}\boldsymbol{I}_S \\ \boldsymbol{I}_S & \boldsymbol{0} \end{bmatrix}}_{\boldsymbol{M}} \\ &\quad \times \underbrace{\begin{bmatrix} \tilde{\boldsymbol{X}}^{k}_{S,:} - \bar{\boldsymbol{X}}_{S,:} \\ \tilde{\boldsymbol{X}}^{k-1}_{S,:} - \bar{\boldsymbol{X}}_{S,:} \end{bmatrix}}_{\boldsymbol{z}^{k-1}} \\ &\quad - \theta_k \begin{bmatrix} \partial g_{\eta_k}(\|\tilde{\boldsymbol{X}}^{k+1}_{S,:}\|_2) \\ \boldsymbol{0} \end{bmatrix}. \end{aligned} \tag{77}$$

There exist a nonsingular matrix $\boldsymbol{T} \in \mathbb{C}^{2|S|\times 2|S|}$ and a diagonal matrix $\boldsymbol{\Lambda_M} \in \mathbb{C}^{2|S|\times 2|S|}$ such that matrix $\boldsymbol{M}$ can be factorized as $\boldsymbol{M} = \boldsymbol{T}\boldsymbol{\Lambda_M}\boldsymbol{T}^{-1}$. The matrices $\boldsymbol{T}$ and $\boldsymbol{\Lambda_M}$ satisfy

$$\|\boldsymbol{\Lambda_M}\|_F = \sqrt{2|S|\hat{\beta}} \le \sqrt{2s\hat{\beta}}, \tag{78}$$

$$\|\boldsymbol{T}\|_F = \sqrt{|S|(2+2\hat{\beta})} \le \sqrt{2s(1+\hat{\beta})}, \tag{79}$$

$$\|\boldsymbol{T}^{-1}\|_F \le \sqrt{\frac{|S|(2+2\hat{\beta})}{4\hat{\beta} - (\hat{\beta}+\phi s - \phi)^2}} \le \sqrt{\frac{2s(1+\hat{\beta})}{4\hat{\beta} - (\hat{\beta}+\phi s-\phi)^2}}. \tag{80}$$

The proof follows the idea in [31] with some modifications according to the problem that we consider.

Before we estimate the recovery error $\|\tilde{\boldsymbol{X}}^{k+1} - \tilde{\boldsymbol{X}}^*\|_F$, we use induction to prove $\partial g_{\eta_t}(\|\tilde{\boldsymbol{X}}^{t+1}_{S,:}\|_2)$ $= \boldsymbol{0}$ for all t satisfying $K_0 \le t \le k\,(k \ge K_0)$.

(i) We prove that $\partial g_{\eta_{K_0}}(\|\tilde{\boldsymbol{X}}^{K_0+1}_{S,:}\|_2) = \boldsymbol{0}$. According to the definition of K_0 (50), it holds that $C_0(c_{\phi s})^k < \underline{\mu_x}/6$ for $\forall\, k \ge K_0$. Based on the assumption of ϵ (47), it follows that $(1+s)\mu_B\epsilon/(1-c_{\phi s}) \le \underline{\mu_x}/6$.

According to (71), when $k = K_0+1$, for $\forall\, i \in S$, we have

$$\begin{aligned} \|\tilde{\boldsymbol{X}}^{K_0+1}_{i,:} - \tilde{\boldsymbol{X}}^*_{i,:}\|_2 \le \|\tilde{\boldsymbol{X}}^{K_0+1} - \tilde{\boldsymbol{X}}^*\|_F \le C_0(c_{\phi s})^{K_0+1} \\ + \frac{(1+s)\mu_B\epsilon}{1-c_{\phi s}} < \frac{\underline{\mu_x}}{3} < \underline{\mu_x}. \end{aligned} \tag{81}$$

Then, we prove that $\|\tilde{\boldsymbol{X}}^{K_0+1}_{i,:}\|_2 > 0, \forall\, i \in S$. If $\|\tilde{\boldsymbol{X}}^{K_0+1}_{i,:}\|_2 = 0$, then we have $\tilde{\boldsymbol{X}}^{K_0+1}_{i,j} = 0, \forall j \in [M]$. Hence, we obtain $\|\tilde{\boldsymbol{X}}^{K_0+1}_{i,:} - \tilde{\boldsymbol{X}}^*_{i,:}\|_2 = \|\tilde{\boldsymbol{X}}^*_{i,:}\|_2 < \underline{\mu_x}$, which contradicts with the assumption $\|\tilde{\boldsymbol{X}}^*_{i,:}\|_2 \ge \underline{\mu_x} > 0$. Therefore, we obtain $\|\tilde{\boldsymbol{X}}^{K_0+1}_{i,:}\|_2 > 0$.

The definition of η_k in (52) implies that $\eta_k = \frac{1}{2\theta_k}$ when $k \ge K_0$. Hence, the univariate proximal operator $\hat{P}_{\theta_k, f_{\eta_k}}(\cdot)$ becomes a hard thresholding function, i.e.,

$$\hat{P}_{\theta_k, f_{\eta_k}}(x) = \begin{cases} 0, & \text{if } |x| \le \theta_k, \\ x, & \text{if } |x| > \theta_k. \end{cases} \tag{82}$$

With (64), (82), and $\|\tilde{\boldsymbol{X}}^{K_0+1}_{i,:}\|_2 > 0$, we obtain $\partial g_{\eta_{K_0}}(\|\tilde{\boldsymbol{X}}^{K_0+1}_{i,:}\|_2) = \boldsymbol{0}$. Hence, we prove that $\partial g_{\eta_{K_0}}(\|\tilde{\boldsymbol{X}}^{K_0+1}_{S,:}\|_2) = \boldsymbol{0}$.

(ii) We assume that $\partial g_{\eta_t}(\|\tilde{\boldsymbol{X}}^{t+1}_{S,:}\|_2) = \boldsymbol{0}$ for $K_0 \le t \le k$. According to (77) and $\boldsymbol{M} = \boldsymbol{T}\boldsymbol{\Lambda_M}\boldsymbol{T}^{-1}$, we have

$$\boldsymbol{z}^t = \boldsymbol{T}\boldsymbol{\Lambda_M}\boldsymbol{T}^{-1}\boldsymbol{z}^{t-1}, \quad K_0 + 1 \le t \le k. \tag{83}$$

Then, we obtain

$$\boldsymbol{z}^k = \boldsymbol{T}(\boldsymbol{\Lambda_M})^{k-K_0}\boldsymbol{T}^{-1}\boldsymbol{z}^{K_0}. \tag{84}$$

By taking norm on both sides of (84), we have

$$\begin{aligned} \|\boldsymbol{z}^k\|_F &\le \|\boldsymbol{T}\|_F \|\boldsymbol{\Lambda_M}\|_F^{k-K_0} \|\boldsymbol{T}^{-1}\|_F \|\boldsymbol{z}^{K_0}\|_F \\ &\le \frac{2s(1+\hat{\beta})\|\boldsymbol{z}^{K_0}\|_F}{\sqrt{4\hat{\beta} - (\hat{\beta}+\phi s-\phi)^2}} \left(\sqrt{2s\hat{\beta}}\right)^{k-K_0}. \end{aligned} \tag{85}$$

Next, we bound $\|\boldsymbol{z}^{K_0}\|_F$. Based on (76), $\beta_{K_0} = 0$ and $\partial g_{\eta_{K_0}}(\|\tilde{\boldsymbol{X}}_{S,:}^{K_0+1}\|_2) = \mathbf{0}$, we have

$$\tilde{\boldsymbol{X}}_{S,:}^{K_0+1} - \bar{\boldsymbol{X}}_{S,:} = (\boldsymbol{I}_S - \boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S})(\tilde{\boldsymbol{X}}_{S,:}^{K_0} - \bar{\boldsymbol{X}}_{S,:}). \tag{86}$$

Due to the definition of ϕ, we know that the elements of matrix $\boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S}$ except the diagonal elements are not larger than ϕ, and the diagonal elements are zero. Thus, we have $\|\boldsymbol{I}_S - \boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S}\|_F \leq \sqrt{|S|(|S|-1)}\phi \leq \sqrt{s(s-1)}\phi$.

By taking norm on both sides of (86), we have

$$\begin{aligned}
&\|\tilde{\boldsymbol{X}}_{S,:}^{K_0+1} - \bar{\boldsymbol{X}}_{S,:}\|_F \\
&\quad \leq \|\boldsymbol{I}_S - \boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S}\|_F \|\tilde{\boldsymbol{X}}_{S,:}^{K_0} - \bar{\boldsymbol{X}}_{S,:}\|_F \\
&\quad \leq \sqrt{s(s-1)}\phi\|\tilde{\boldsymbol{X}}_{S,:}^{K_0} - \bar{\boldsymbol{X}}_{S,:}\|_F \\
&\quad \leq (2s-1)\phi\|\tilde{\boldsymbol{X}}_{S,:}^{K_0} - \bar{\boldsymbol{X}}_{S,:}\|_F \leq \|\tilde{\boldsymbol{X}}_{S,:}^{K_0} - \bar{\boldsymbol{X}}_{S,:}\|_F.
\end{aligned} \tag{87}$$

Recalling that $\boldsymbol{z}^{K_0} = [(\tilde{\boldsymbol{X}}_{S,:}^{K_0+1} - \bar{\boldsymbol{X}}_{S,:})^{\mathrm{T}}, (\tilde{\boldsymbol{X}}_{S,:}^{K_0} - \bar{\boldsymbol{X}}_{S,:})^{\mathrm{T}}]^{\mathrm{T}}$, we have

$$\begin{aligned}
\|\boldsymbol{z}^{K_0}\|_F &\leq 2\|\tilde{\boldsymbol{X}}_{S,:}^{K_0} - \bar{\boldsymbol{X}}_{S,:}\|_F \\
&\leq 2\|\tilde{\boldsymbol{X}}_{S,:}^{K_0} - \tilde{\boldsymbol{X}}_{S,:}^{*}\|_F + 2\|\delta\tilde{\boldsymbol{X}}_{S,:}\|_F \\
&\leq 2\sqrt{s}\mu_x(c_{\phi s})^{K_0} + \frac{2(1+s)\mu_B\epsilon}{1-c_{\phi s}} + \frac{2\sqrt{s}\mu_B}{1+\phi-\phi s}\epsilon \\
&\leq 2\sqrt{s}\mu_x(c_{\phi s})^{K_0} + \frac{2(1+s+\sqrt{s})\mu_B\epsilon}{1-c_{\phi s}} \\
&\leq 2\sqrt{s}\mu_x(c_{\phi s})^{K_0} + 2\sqrt{s}\mu_x(c_{\phi s})^{K_0} = 4\sqrt{s}\mu_x(c_{\phi s})^{K_0},
\end{aligned} \tag{88}$$

where the last inequality follows from (47).

Combining with (85), we obtain

$$\begin{aligned}
\|\tilde{\boldsymbol{X}}_{S,:}^{k+1} - \bar{\boldsymbol{X}}_{S,:}\|_F &\leq \|\boldsymbol{z}^k\|_F \\
&\leq \frac{8\mu_x s\sqrt{s}(1+\hat{\beta})(c_{\phi s})^{K_0}}{\sqrt{4\hat{\beta} - (\hat{\beta} + \phi s - \phi)^2}} \left(\sqrt{2s\hat{\beta}}\right)^{k-K_0} \\
&\leq C_0(c_{\phi s})^{K_0+1}\left(\sqrt{2s\hat{\beta}}\right)^{k-K_0}.
\end{aligned} \tag{89}$$

Subsequently, by defining $\hat{\boldsymbol{X}}_{S,:}^{k+1} = \tilde{\boldsymbol{X}}_{S,:}^{k+1} - \boldsymbol{B}_{S,:}\tilde{\boldsymbol{S}}_{:,S}(\tilde{\boldsymbol{X}}_{S,:}^{k+1} - \bar{\boldsymbol{X}}_{S,:}) + \beta_{k+1}(\tilde{\boldsymbol{X}}_{S,:}^{k+1} - \tilde{\boldsymbol{X}}_{S,:}^{k})$ and $\hat{\boldsymbol{z}}^k = [(\hat{\boldsymbol{X}}_{S,:}^{k+1} - \bar{\boldsymbol{X}}_{S,:})^{\mathrm{T}}, (\tilde{\boldsymbol{X}}_{S,:}^{k+1} - \bar{\boldsymbol{X}}_{S,:})^{\mathrm{T}}]^{\mathrm{T}}$, we have $\hat{\boldsymbol{z}}^k = \boldsymbol{M}\boldsymbol{z}^k$ and $\tilde{\boldsymbol{X}}_{S,:}^{k+2} = P_{\theta_{k+1}, f_{\eta_{k+1}}}(\hat{\boldsymbol{X}}_{S,:}^{k+1})$. Following the same idea of proving (89), we obtain

$$\|\hat{\boldsymbol{X}}_{S,:}^{k+1} - \bar{\boldsymbol{X}}_{S,:}\|_F \leq \|\hat{\boldsymbol{z}}^k\|_F \leq C_0(c_{\phi s})^{K_0+1}\left(\sqrt{2s\hat{\beta}}\right)^{k+1-K_0}. \tag{90}$$

Since the inequality $\sqrt{2s\hat{\beta}} = 1 - \sqrt{1-c_{\phi s}} \leq c_{\phi s}$ holds when $c_{\phi s} < 1$, we have

$$\|\hat{\boldsymbol{X}}_{S,:}^{k+1} - \bar{\boldsymbol{X}}_{S,:}\|_F \leq C_0(c_{\phi s})^{k+2} < \underline{\mu_x}/6. \tag{91}$$

Recalling that $\bar{\boldsymbol{X}}_{S,:} = \tilde{\boldsymbol{X}}_{S,:}^{*} + \delta\tilde{\boldsymbol{X}}_{S,:}$, for $\forall\, i \in S$, we obtain

$$\begin{aligned}
\|\hat{\boldsymbol{X}}_{i,:}^{k+1} - \tilde{\boldsymbol{X}}_{i,:}^{*}\|_2 &\leq \|\hat{\boldsymbol{X}}_{S,:}^{k+1} - \tilde{\boldsymbol{X}}_{S,:}^{*}\|_F \leq \|\hat{\boldsymbol{X}}_{S,:}^{k+1} - \bar{\boldsymbol{X}}_{S,:}\|_F \\
&\quad + \|\delta\tilde{\boldsymbol{X}}_{S,:}\|_F < \underline{\mu_x}/6 + \underline{\mu_x}/6 = \underline{\mu_x}/3.
\end{aligned} \tag{92}$$

Hence, for $\forall\, i \in S$, we have

$$\|\hat{\boldsymbol{X}}_{i,:}^{k+1}\|_2 \geq \|\tilde{\boldsymbol{X}}_{i,:}^{*}\|_2 - \|\hat{\boldsymbol{X}}_{i,:}^{k+1} - \tilde{\boldsymbol{X}}_{i,:}^{*}\|_2 > \underline{\mu_x} - \underline{\mu_x}/3 = 2\underline{\mu_x}/3. \tag{93}$$

With (73), we obtain

$$\|\delta\tilde{\boldsymbol{X}}_{S,:}\|_F \leq \frac{\sqrt{s}\mu_B\epsilon}{1+\phi-\phi s} \leq \frac{\sqrt{s}\mu_B\epsilon}{1+\phi-2\phi s} \leq \frac{\underline{\mu_x}}{6}\frac{\sqrt{s}}{s+1} \leq \frac{\underline{\mu_x}}{6}. \tag{94}$$

By the definition of $\hat{\beta}$, we have

$$\begin{aligned}
\|\tilde{\boldsymbol{X}}_{i,:}^{k+1} - \bar{\boldsymbol{X}}_{i,:}^{*}\|_2 &\leq \|\tilde{\boldsymbol{X}}_{S,:}^{k+1} - \bar{\boldsymbol{X}}_{S,:}\|_F \\
&\leq C_0(c_{\phi s})^{K_0+1}\left(\sqrt{2s\hat{\beta}}\right)^{k-K_0} \\
&\leq C_0(c_{\phi s})^{k+1} \leq \underline{\mu_x}/6.
\end{aligned}$$

Thus, we obtain

$$\begin{aligned}
\|\tilde{\boldsymbol{X}}_{i,:}^{k+1} - \tilde{\boldsymbol{X}}_{i,:}^{*}\|_2 &\leq \|\tilde{\boldsymbol{X}}_{i,:}^{k+1} - \bar{\boldsymbol{X}}_{i,:}^{*}\|_2 + \|\delta\tilde{\boldsymbol{X}}_{i,:}^{k+1}\|_2 \\
&\leq \underline{\mu_x}/6 + \underline{\mu_x}/6 = \underline{\mu_x}/3.
\end{aligned} \tag{95}$$

On the other hand, according to the definition of θ_k, we have

$$\begin{aligned}
\theta_{k+1} &= \phi \sup_{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)} \|\tilde{\boldsymbol{X}}^{k+1} - \tilde{\boldsymbol{X}}^{*}\|_{2,1} + \mu_B\epsilon \\
&= \phi \sup_{(\tilde{\boldsymbol{X}}^*, \tilde{\boldsymbol{Z}}) \in \mathcal{X}(\underline{\mu_x}, \mu_x, s, \epsilon)} \sum_{i=1}^{|S|} \|\tilde{\boldsymbol{X}}_{i,:}^{k+1} - \tilde{\boldsymbol{X}}_{i,:}^{*}\|_2 + \mu_B\epsilon \\
&\leq \phi s\underline{\mu_x}/3 + \mu_B\epsilon \leq \underline{\mu_x}/3 + \underline{\mu_x}/3 = 2\underline{\mu_x}/3.
\end{aligned} \tag{96}$$

Because of $\tilde{\boldsymbol{X}}_{S,:}^{k+2} = P_{\theta_{k+1}, f_{\eta_{k+1}}}(\hat{\boldsymbol{X}}_{S,:}^{k+1})$ and the inequality $\|\hat{\boldsymbol{X}}_{i,:}^{k+1}\|_2 > 2\underline{\mu_x}/3 \geq \theta_{k+1}$, we obtain $\tilde{\boldsymbol{X}}_{i,:}^{k+2} = \hat{\boldsymbol{X}}_{i,:}^{k+1}, \forall\, i \in S$. Thus, we prove that $\|\tilde{\boldsymbol{X}}_{i,:}^{k+2}\|_2 > 0$. Finally, we can obtain $\partial g_{\eta_{k+1}}(\|\tilde{\boldsymbol{X}}_{S,:}^{k+2}\|_2) = \mathbf{0}$.

We have proved $\partial g_{\eta_t}(\|\tilde{\boldsymbol{X}}_{S,:}^{t+1}\|_2) = \mathbf{0}$ for all t satisfying $K_0 \leq t \leq k\,(k \geq K_0)$ by induction. According to (89), we have

$$\begin{aligned}
\|\tilde{\boldsymbol{X}}^{k+1} - \tilde{\boldsymbol{X}}^{*}\|_F &\leq \|\tilde{\boldsymbol{X}}_{S,:}^{k+1} - \bar{\boldsymbol{X}}_{S,:}\|_F + \|\delta\tilde{\boldsymbol{X}}_{S,:}\|_F \\
&\leq C_0(c_{\phi s})^{K_0+1}\left(1 - \sqrt{1-c_{\phi s}}\right)^{k-K_0} \\
&\quad + \frac{s\mu_B}{1+\phi-\phi s}\epsilon, \\
&\quad \forall\, k \geq K_0 + 1.
\end{aligned} \tag{97}$$

References

[1] Y. Zou, Y. Zhou, Y. Shi, and X. Chen, "Learning proximal operator methods for massive connectivity in IoT networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.

[2] K. Sharma and X. Wang, "Toward massive machine type communications in ultra-dense cellular IoT networks: Current issues and machine learning-assisted solutions," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 426–471, 1st Quart., 2019.

[3] L. Liu, E. G. Larsson, W. Yu, P. Popovski, C. Stefanovic, and E. de Carvalh, "Sparse signal processing for grant-free massive connectivity: A future paradigm for random access protocols in the Internet of Things," *IEEE Signal Process. Mag.*, vol. 35, no. 5, pp. 88–99, Sep. 2018.

[4] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[5] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 5302–5316, Nov. 2009.

[6] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3042–3054, Jun. 2010.

[7] Z. Qin, K. Scheinberg, and D. Goldfarb, "Efficient block-coordinate descent algorithms for the group lasso," *Math. Program. Comput.*, vol. 5, no. 2, pp. 143–169, Jun. 2013.

[8] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Roy. Stat. Soc. B, Stat. Methodology*, vol. 68, no. 1, pp. 49–67, Feb. 2006.

[9] T. Jiang, Y. Shi, J. Zhang, and K. B. Letaief, "Joint activity detection and channel estimation for IoT networks: Phase transition and computation-estimation tradeoff," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6212–6225, Aug. 2019.

[10] X. Shao, X. Chen, and R. Jia, "A dimension reduction-based joint activity detection and channel estimation algorithm for massive access," *IEEE Trans. Signal Process.*, vol. 68, no. 2, pp. 420–435, Dec. 2019.

[11] X. Shao, X. Chen, C. Zhong, and Z. Zhang, "Exploiting simultaneous low-rank and sparsity in delay-angular domain for millimeter-wave/terahertz wideband massive access," *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2336–2351, Apr. 2022.

[12] Q. He, T. Q. S. Quek, Z. Chen, Q. Zhang, and S. Li, "Compressive channel estimation and multi-user detection in C-RAN with low-complexity methods," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 3931–3944, Jun. 2018.

[13] 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, Jun. 2018.

[14] Z. Chen, F. Sohrabi, and W. Yu, "Multi-cell sparse activity detection for massive random access: Massive MIMO versus cooperative MIMO," *IEEE Trans. Wireless Commun.*, vol. 18, no. 8, pp. 4060–4074, Aug. 2019.

[15] S. Xia, Y. Shi, Y. Zhou, and X. Yuan, "Reconfigurable intelligent surface for massive connectivity: Joint activity detection and channel estimation," *IEEE Trans. Signal Process.*, vol. 69, pp. 5693–5707, 2021.

[16] S. Rangan, P. Schniter, and A. K. Fletcher, "Vector approximate message passing," *IEEE Trans. Inf. Theory*, vol. 65, no. 10, pp. 6664–6684, Oct. 2019.

[17] S. Rangan, P. Schniter, A. K. Fletcher, and S. Sarkar, "On the convergence of approximate message passing with arbitrary matrices," *IEEE Trans. Inf. Theory*, vol. 65, no. 9, pp. 5339–5351, Sep. 2019.

[18] Y. C. Eldar, A. Goldsmith, D. Gündüz, and H. V. Poor, *Machine Learning and Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2022.

[19] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing," *IEEE Signal Process. Mag.*, vol. 38, no. 2, pp. 18–44, Mar. 2021.

[20] J. Scarlett, R. Heckel, M. R. D. Rodrigues, P. Hand, and Y. C. Eldar, "Theoretical perspectives on deep learning methods in inverse problems," *IEEE J. Sel. Areas Inf. Theory*, vol. 3, no. 3, pp. 433–453, Sep. 2022.

[21] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, Jun. 2010, pp. 399–406.

[22] M. Borgerding, P. Schniter, and S. Rangan, "AMP-inspired deep networks for sparse linear inverse problems," *IEEE Trans. Signal Process.*, vol. 65, no. 16, pp. 4293–4308, Aug. 2017.

[23] X. Chen, J. Liu, Z. Wang, and W. Yin, "Theoretical linear convergence of unfolded ISTA and its practical weights and thresholds," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2018, pp. 9061–9071.

[24] J. Liu, X. Chen, Z. Wang, and W. Yin, "ALISTA: Analytic weights are as good as learned weights in LISTA," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–33.

[25] Y. Shi, H. Choi, Y. Shi, and Y. Zhou, "Algorithm unrolling for massive access via deep neural networks with theoretical guarantee," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 945–959, Feb. 2022.

[26] Y. Cui, S. Li, and W. Zhang, "Jointly sparse signal recovery and support recovery via deep learning with applications in MIMO-based grant-free random access," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 3, pp. 788–803, Mar. 2021.

[27] J. Johnston and X. Wang, "Model-based deep learning for joint activity detection and channel estimation in massive and sporadic connectivity," *IEEE Trans. Wireless Commun.*, vol. 21, no. 11, pp. 9806–9817, Nov. 2022.

[28] W. Zhu, M. Tao, X. Yuan, and Y. Guan, "Deep-learned approximate message passing for asynchronous massive connectivity," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5434–5448, Aug. 2021.

[29] C. Yang, Y. Gu, B. Chen, H. Ma, and H. C. So, "Learning proximal operator methods for nonconvex sparse recovery with theoretical guarantee," *IEEE Trans. Signal Process.*, vol. 68, pp. 5244–5259, 2020.

[30] X. Shao, X. Chen, Y. Qiang, C. Zhong, and Z. Zhang, "Feature-aided adaptive-tuning deep learning for massive device detection," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 7, pp. 1899–1914, Jul. 2021.

[31] X. Chen, J. Liu, Z. Wang, and W. Yin, "Hyperparameter tuning is all you need for LISTA," in *Proc. Neural Inf. Process. Syst. (NeurIPS)*, vol. 34, 2021, pp. 11678–11689.

[32] A. Rajoriya and R. Budhiraja, "Joint AMP-SBL algorithms for device activity detection and channel estimation in massive MIMO mMTC systems," *IEEE Trans. Commun.*, vol. 71, no. 4, pp. 2136–2152, Apr. 2023.

[33] X. Shao, X. Chen, C. Zhong, J. Zhao, and Z. Zhang, "A unified design of massive access for cellular Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3934–3947, Apr. 2019.

[34] 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.

[35] Z. Chen, F. Sohrabi, and W. Yu, "Sparse activity detection for massive connectivity," *IEEE Trans. Signal Process.*, vol. 66, no. 7, pp. 1890–1904, Apr. 2018.

[36] Y. Jiang, J. Su, Y. Shi, and B. Houska, "Distributed optimization for massive connectivity," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1412–1416, Sep. 2020.

[37] C.-H. Zhang, "Nearly unbiased variable selection under minimax concave penalty," *Ann. Statist.*, vol. 38, no. 2, pp. 894–942, 2010.

[38] P. Breheny and J. Huang, "Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors," *Statist. Comput.*, vol. 25, no. 2, pp. 173–187, Mar. 2015.

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

[40] A. Beck, *First-order Methods in Optimization*. Philadelphia, PA, USA: SIAM, 2017.

[41] P. Breheny and J. Huang, "Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection," *Ann. Appl. Statist.*, vol. 5, no. 1, p. 232, Mar. 2011.

[42] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods," *USSR Comput. Math. Math. Phys.*, vol. 4, no. 5, pp. 1–17, Jan. 1964.

[43] C. Lu, H. Li, and Z. Lin, "Optimized projections for compressed sensing via direct mutual coherence minimization," *Signal Process.*, vol. 151, pp. 45–55, Oct. 2018.

[44] D. G. Luenberger and Y. Ye, *Linear and Nonlinear Programming*, vol. 2. New York, NY, USA: Springer, 1984.

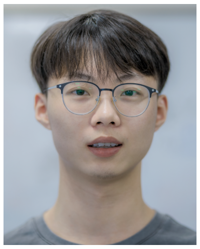
[45] D. Chu, "Polyphase codes with good periodic correlation properties (corresp.)," *IEEE Trans. Inf. Theory*, vol. IT-18, no. 4, pp. 531–532, Jul. 1972.

[46] J. H. I. de Souza and T. Abrão, "Deep learning-based activity detection for grant-free random access," *IEEE Syst. J.*, vol. 17, no. 1, pp. 940–951, Mar. 2023.

[47] 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, Jan. 2009.

[48] C. Steffens, M. Pesavento, and M. E. Pfetsch, "A compact formulation for the $\ell_{2,1}$ mixed-norm minimization problem," *IEEE Trans. Signal Process.*, vol. 66, no. 6, pp. 1483–1497, Mar. 2018.

[49] G. H. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, MD, USA: JHU Press, 2013.



Yinan Zou (Graduate Student Member, IEEE) received the B.E. degree in electronic information engineering from Chongqing University, Chongqing, China, in 2020, and the M.Eng. degree in information and communication engineering from ShanghaiTech University, Shanghai, China, in 2023.

Yong Zhou (Senior Member, IEEE) received the B.Sc. and M.Eng. degrees from Shandong University, Jinan, China, in 2008 and 2011, respectively, and the Ph.D. degree from the University of Waterloo, Waterloo, ON, Canada, in 2015. From November 2015 to January 2018, he worked as a Post-Doctoral Researcher Fellow with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, Canada. Since March 2018, he has been with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China, where he is currently a Tenured Associate Professor. His research interests include 6G communications, edge intelligence, and the Internet of Things. He was the Track Co-Chair of IEEE VTC 2020 (Fall) and IEEE VTC 2023 (Spring) and the General Co-Chair of the IEEE ICC 2022 Workshop on Edge Artificial Intelligence for 6G. He serves as an Associate Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY.

Xu Chen (Senior Member, IEEE) received the Ph.D. degree in information engineering from The Chinese University of Hong Kong in 2012. He worked as a Post-Doctoral Research Associate with Arizona State University, Tempe, AZ, USA, from 2012 to 2014, and a Humboldt Scholar Fellow with the Institute of Computer Science, University of Göttingen, Germany, from 2014 to 2016. He is currently a Full Professor with Sun Yat-sen University, Guangzhou, China, the Director of the Institute of Advanced Networking and Computing Systems, and the Vice Director of the National Engineering Research Laboratory of Digital Homes. He received the prestigious Humboldt Research Fellowship Awarded by the Alexander von Humboldt Foundation of Germany, Hong Kong Young Scientist Award, the IEEE Communication Society Asia-Pacific Outstanding Young Researcher Award and Outstanding Paper Award, the IEEE Communication Society Young Professional Best Paper Award, the IEEE Computer Society Best Paper Awards Runner-Up, the IEEE ICC Best Paper Award, the IEEE ISI Honorable Mention Award, the Best Paper Runner-Up Awards of IEEE INFOCOM and IEEE INTERNET OF THINGS JOURNAL. He is an Area Editor of IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY, an Associate Editor of IEEE TRANSACTIONS WIRELESS COMMUNICATIONS, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (JSAC) Series on Network Softwarization and Enablers.

Yonina C. Eldar (Fellow, IEEE) received the B.Sc. degree in physics and the B.Sc. degree in electrical engineering from Tel Aviv University and the Ph.D. degree in electrical engineering and computer science from MIT in 2002. She was a Visiting Professor in Stanford. She was a Horev Fellow of the Leaders in Science and Technology Program, Technion, and an Alon Fellow. She is currently a Professor with the Department of Mathematics and Computer Science, Weizmann Institute of Science, Rehovot, Israel, where she heads the Center for Biomedical Engineering and Signal Processing and holds the Dorothy and Patrick Gorman Professorial Chair. She is also a Visiting Professor with MIT, a Visiting Scientist with the Broad Institute, and an Adjunct Professor with Duke University. She is a member of Israel Academy of Sciences and Humanities and a EURASIP Fellow. She has received many awards for excellence in research and teaching, including the IEEE Signal Processing Society Technical Achievement Award in 2013, the IEEE/AESS Fred Nathanson Memorial Radar Award in 2014, and the IEEE Kiyo Tomiyasu Award in 2016. She received the Michael Bruno Memorial Award from the Rothschild Foundation, the Weizmann Prize for Exact Sciences, the Wolf Foundation Krill Prize for Excellence in Scientific Research, the Henry Taub Prize for Excellence in Research (twice), the Hershel Rich Innovation Award (three times), and the Award for Women with Distinguished Contributions. She received several best paper awards and best demo awards together with her research students and colleagues, was selected as one of the 50 most influential women in Israel, and was a member of Israel Committee for Higher Education. She is the Editor-in-Chief of *Foundations and Trends in Signal Processing*, a member of several IEEE technical committees and award committees, and heads the Committee for Promoting Gender Fairness in Higher Education Institutions, Israel.