Iterative-Weighted Thresholding Method for Group-Sparsity-Constrained Optimization with Applications

Lanfan Jiang, Zilin Huang, Yu Chen, and Wenxing Zhu

***Abstract*—Taking advantage of the natural grouping structure inside data, group sparse optimization can effectively improve the efficiency and stability of high-dimensional data analysis, and it has wide applications in a variety of fields such as machine learning, signal processing, and bioinformatics. Although there has been a lot of progress, it is still a challenge to construct a group sparse inducing function with good properties and to identify significant groups. This paper aims to address the group sparsity constrained minimization problem. We convert the problem to an equivalent weighted $\ell_{p,q}$-norm $(p > 0, 0 < q \leq 1)$ constrained optimization model, instead of its relaxation or approximation problem. Then by applying the proximal gradient method, a solution method with theoretical convergence analysis is developed. Moreover, based on the properties proved in the Lagrangian dual framework, the homotopy technique is employed to cope with the parameter tuning task and to ensure that the output of the proposed homotopy algorithm is an L-stationary point of the original problem. The proposed weighted framework, with the central idea of identifying important groups, is compatible with a wide range of support set identification strategies, which can better meet the needs of different applications and improve the robustness of the model in practice. Both simulated and real data experiments demonstrate the superiority of the proposed method in terms of group feature selection accuracy and computational efficiency. Extensive experimental results in application areas such as compressed sensing, image recognition, and classifier design show that our method has great potential in a wide range of applications. Our codes will be available at https://github.com/jianglanfan/HIWT-GSC.**

***Index Terms*—Group sparse, sparse optimization, iterative weighted thresholding, homotopy, proximal gradient, non-convex optimization.**

I. Introduction

In an increasingly digitized world, the rapid growth of feature dimensions brings new challenges to data processing and analysis. Sparse structure, as an important low-dimensional structure in high-dimensional data, is widely available in a large number of practical applications. By utilizing the sparse structure of data, sparse optimization models perform dimension reduction and feature selection, thereby achieving the purpose of reducing complexity, enhancing interpretability, and improving efficiency of the task. Therefore, sparse optimization models have become a popular modeling tool in the context of high-dimensional data, and have demonstrated their power in many fields such as image processing [1], [2], computer vision [3], [4], and machine learning [5], [6].

In recent years, optimization problems emerging from many application areas have data with more structural information than just sparsity. For example, in gene expression analysis, genes belonging to the same biological pathway can be considered as a group [7]. In image processing, the color space representation of a pixel can be thought of as a group [8]. In deep neural network compression, all of the output weights of a neuron in a network can be considered as a group [9], [10]. Some literature in statistics and machine learning [11]–[15] has shown that when data have a certain group structure, group sparse optimization models reduce the degrees of freedom in problem-solving, leading to better solutions with fewer measurement requirements and lower computational complexity compared to standard sparse optimization models. Therefore, group sparsity, as an important constraint, has been widely studied and applied in the past decade[16]–[20].

Given a function $f : \mathbb{R}^n \to \mathbb{R}$, in this work, we aim to optimize the following group-sparsity constrained minimization problem with non-overlapping groups:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) \quad \text{s.t. } \|\boldsymbol{x}\|_{p,0} \leq s, \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is a coefficient vector of features with group structure which is to be estimated, and $\|\boldsymbol{x}\|_{p,0} (p > 0)$ counts the number of non-zero groups in $\boldsymbol{x}$.

The group structure partitions n features into N groups with a pre-determined non-overlapping index set $\mathbb{G} = \{\mathbb{G}_1, \ldots, \mathbb{G}_N\}$, hence in problem (1), $0 < s \leq N$. The $\ell_{p,q}$ $(p > 0)$ norm of $\boldsymbol{x}$ is defined as

$$\|\boldsymbol{x}\|_{p,q} = \begin{cases} \left(\sum_{i=1}^N \|\boldsymbol{x}_{\mathbb{G}_i}\|_p^q\right)^{\frac{1}{q}}, & q > 0; \\ |\{i : \|\boldsymbol{x}_{\mathbb{G}_i}\|_p \neq 0\}|, & q = 0; \\ \max_i\{\|\boldsymbol{x}_{\mathbb{G}_i}\|_p\}, & q = \infty. \end{cases} \tag{2}$$

Actually, not all features are informative. Problem (1) aims to identify at most s relevant and informative groups of features in the model with respect to minimizing the function

This work was supported in part by the National Key R&D Program of China under Grant 2023YFA1011302, the National Natural Science Foundation of China under Grant 62174033, and the Central Guidance on Local Science and Technology Development Fund of Fujian Province (No. 2023L3003). Additional support was provided by the Educational Research Program for Young and Middle-aged Teachers in Fujian Province under Grant JAT210023. *(Corresponding author: Wenxing Zhu).*

Lanfan Jiang and Yu Chen are with the College of Computer and Data Science, Fuzhou University, Fuzhou 350116, China (e-mail: jianglanfan@foxmail.com, chenyu.fzu@foxmail.com).

Zilin Huang is with the School of Computer and Big Data, Minjiang University, Fuzhou 350108, China (e-mail: zlhuangoptimal@163.com).

Wenxing Zhu is with the Center for Discrete Mathematics and Theoretical Computer Science, Fuzhou University, Fuzhou 350116, China (e-mail: wxzhu@fzu.edu.cn).

$f(\boldsymbol{x})$. Since $\|\cdot\|_{p,0}$ is a non-convex and discontinuous function, the optimization problem associated with the $\ell_{p,0}$-norm is generally hard to solve. To this end, many plausible methods have been proposed for $\ell_{p,0}$-norm based problems. We briefly review the methods that directly motivated our research as follows.

Relaxation is a widely used technique in sparse optimization. This technique relaxes the discontinuous $\ell_{p,0}$ function into a more tractable convex or non-convex group sparsity-inducing function, denoted by $\psi(\boldsymbol{x})$. A common approach is the convex relaxation of the $\ell_{p,0}$ regularization to the $\ell_{2,1}$ regularization, often referred to as group least absolute shrinkage and selection operator (GLASSO) [21]. The regularization term of the GLASSO is defined by

$$\psi_\lambda(\boldsymbol{x}) = \lambda \sum_{i=1}^{N} \|\boldsymbol{x}_{\mathbb{G}_i}\|_2 \,, \tag{3}$$

where $\lambda > 0$ is the regularization parameter. As can be seen from (3), similar to LASSO, GLASSO imposes the same degree of penalty on all groups, resulting in a biased solution.

To overcome this drawback, some non-convex methods, such as group smoothly clipped absolute deviation (GSCAD) [22] and group minimax concave penalty (GMCP) [7], [23], employ piecewise sparsity-inducing functions to penalize different groups with varying degrees according to their magnitude, and consequently possess good consistency in group selection [23]. In a similar vein, researchers have proposed the idea of partial regularization [24]–[26]. Based on such an idea, Feng et al. [27] addressed the block sparse recovery problem by using a less biased group sparsity-inducing function

$$\psi_\lambda(\boldsymbol{x}) = \lambda \sum_{i=s+1}^{N} \left\|\boldsymbol{x}^{\downarrow}_{\mathbb{G}_i}\right\|_2 \,, \tag{4}$$

where $\left\|\boldsymbol{x}^{\downarrow}_{\mathbb{G}_1}\right\|_2 \geq \left\|\boldsymbol{x}^{\downarrow}_{\mathbb{G}_2}\right\|_2 \geq \cdots \geq \left\|\boldsymbol{x}^{\downarrow}_{\mathbb{G}_N}\right\|_2$ are arranged in descending order according to the ℓ_2-norm magnitude of each group. Equation (4) implies that the leading s group entries with the largest magnitude are not penalized, which effectively neutralizes the solution bias in group LASSO. Similar work in recent years can be found in [28]–[30].

The above analyses of the relaxation methods suggest that different groups should be penalized to different degrees. In light of this, it is of great importance to **construct surrogate functions with good properties,** such as group selection consistent, for the $\ell_{p,0}$ function.

Another common approach is to solve the original problem directly, where the key issue is identifying the correct support set. During the identification process, different methods favor different strategies, resulting in different support sets. For example, for the purpose of residual reduction, greedy methods, such as the group orthogonal matching pursuit (GOMP) [31], usually identify contributing features step by step and finally construct the support set of the solution. On the other hand, methods such as iterative hard thresholding (IHT) iteratively perform a gradient descent operation, and then project the resulting vector onto the feasibility set, or take a hard thresholding step to determine the support set. A representative is the group primal-dual active set with continuation (GPDASC) method [32], which solves group sparse regularization problems based on the $\ell_{2,0}$-norm. Another recent analogous study can be referred to the subspace Newton method for sparse group ℓ_0 optimization problem (SNSG) [33], which identifies a support set after two-step projection in each iteration and applies the Newton method within this subspace to search for an improved iteration point.

In general, it is crucial but not straightforward to identify a right support set. In fact, the best support set identification strategies for different applications are often different. For this reason, **a unified framework that is compatible with a wide range of identification strategies is worth studying.**

In summary, there are two major challenges of the group-sparsity optimization problem (1). The first issue is to construct appropriate surrogate functions for the $\ell_{p,0}$-norm based constraints, and the other issue is to identify "informative" groups in real-world problems. Therefore, a general and effective model is required, which not only ensures an accurate solution but also encompasses a wider range of strategies for identifying supporting sets.

This paper aims to address the above issues by proposing a more general group-sparsity framework, referred to as IWT-GSC (Iterative Weighted Thresholding method for Group-Sparsity Constrained problem). It extends the work in [34], [35]. The method proposed in [34] is for variable selection based on the ℓ_1-norm and uses a monotone line search. While, this work is for group selection based on the more general $\ell_{p,q}$-norm and uses a non-monotone line search. The considered problem is more general, moreover, the proposed algorithm is more efficient due to the non-monotone line search [36]. In addition, the method in [35] focuses on group sparse recovery from underdetermined linear systems, and introduces an adaptively updated group threshold hyperparameter into the model to assist in support set detection. In comparison, this work aims to optimize a general group-sparsity constrained problem and allows for a wide range of support set identification strategies, which means we can customize the support set identification strategies to better meet the needs of different applications and to improve the robustness of the model.

The main contributions are summarized as follows:

(i) We convert the original group-sparsity constrained problem (1) to an **equivalent** weighted $\ell_{p,q}$-norm ($p > 0, 0 < q \leq 1$) constrained optimization model instead of its relaxation or approximation problem. In the proposed weighted model, we assign "0-1" weights to different groups to cope with the "biased solution" issue. Moreover, moving beyond single fixed support group identification strategy, our weighted framework actually **allows for a wide range of strategies** to better fit in different applications. Dualizing the $\ell_{p,q}$-norm constraint, we prove that problem (1) has the strong duality property. Applying the proximal gradient method to approximately solve the Lagrangian problem, we present closed-form solutions of a sub-problem for some specific values of p and q.

(ii) We design an efficient and effective IWT-GSC algorithmic framework for solving the proposed Lagrangian problem. The marriage of the Barzilai-Borwein (BB) method and non-monotone line search produces iteratively updated step-sizes which help to speed up the convergence of the IWT-GSC algorithm. Furthermore, to get rid of the tedious but crucial task of parameter tuning and to further enhance the proposed algorithm, we apply the homotopy technique to the IWT-GSC algorithm, which is called HIWT-GSC.

(iii) We demonstrate that the solution sequence $\{\boldsymbol{x}^k\}$ generated by IWT-GSC converges under some mild conditions, and the output of the HIWT-GSC algorithm is an L-stationary point of problem (1).

(iv) In addition to a series of experiments on simulated data, we apply the algorithm to application areas such as compressed sensing, image classification, and classifier design. Extensive experimental results show that our algorithm is reliable in achieving superior selection accuracy of features and provides an accurate solution with high efficiency, compared to state-of-the-art group sparse optimization methods.

The rest of this paper is organized as follows. Section II provides the preliminaries. Section III reformulates the proposed equivalent problem into a problem with a simple constraint via the Lagrangian relaxation, and derives closed-form solutions for some specific values of p and q when applying the proximal gradient method. Section IV explores the IWT-GSC framework and provides convergence analysis. Section V gives the homotopy algorithm HIWT-GSC and elaborates some implementation details. Section VI exhibits a series of experimental results. Finally, Section VII concludes this paper. Due to space limitations, the proofs of the lemmas and theorems are not presented in the main body of this paper, but are provided in the Supplementary Material.

II. Preliminaries

In this section, we provide preliminaries, including notations, definitions, and assumptions.

A. Notations

Throughout this paper, we denote vectors, matrices and sets by lowercase bold letters $\boldsymbol{x}$, uppercase bold letters $\mathbf{A}$, and blackboard bold uppercase letters $\mathbb{S}$, respectively. We use $\boldsymbol{x}_i$ (respectively $\mathbf{A}_i$) to represent the i-th entry (respectively column) of $\boldsymbol{x}$ (respectively $\mathbf{A}$). Unless otherwise stated, we assume that all vectors are column vectors.

Suppose $\boldsymbol{x} \in \mathbb{R}^n$ has a pre-specified non-overlapping group structure with partition $\mathbb{G} = \{\mathbb{G}_1, \ldots, \mathbb{G}_N\}$, where $\mathbb{G}_i$ denote the index set corresponding to the i-th group. Then $\boldsymbol{x}_{\mathbb{G}_i} = (\boldsymbol{x}_j, j \in \mathbb{G}_i)^T$ is the subvector of $\boldsymbol{x}$ indexed by $\mathbb{G}_i$, $\boldsymbol{x} = (\boldsymbol{x}_{\mathbb{G}_1}^T, \ldots, \boldsymbol{x}_{\mathbb{G}_N}^T)^T$ and $\boldsymbol{x}_{\mathbb{G}_i} \cap \boldsymbol{x}_{\mathbb{G}_j} = \varnothing$, where $1 \leq |\mathbb{G}_i| = N_i \leq n$ and $\sum_{i=1}^N |\mathbb{G}_i| = n$. Likewise, $\mathbf{A}_{\mathbb{G}_i} = (\mathbf{A}_j, j \in \mathbb{G}_i)$ is the submatrix of $\mathbf{A}$ indexed by $\mathbb{G}_i$. For convenience, we give the other notations in TABLE I.

B. The definition of soft-thresholding operator

Definition 1: Given parameters λ and L, the soft-thresholding operator $\text{soft}_{\frac{\lambda}{L},p}(\cdot)$ is defined as:

TABLE I
Notations used in this paper

Notation	Description
$\mathbb{S}(\boldsymbol{x})$	$\mathbb{S}(\boldsymbol{x}) = \{i : \boldsymbol{x}_i \neq 0\}$
$\mathbb{S}^c(\boldsymbol{x})$	Complement of $\mathbb{S}(\boldsymbol{x})$, i.e., $\mathbb{S}^c(\boldsymbol{x}) = \{i : \boldsymbol{x}_i = 0\}$
$\mathbb{S}_{\mathbb{G}}(\boldsymbol{x})$	$\mathbb{S}_{\mathbb{G}}(\boldsymbol{x}) = \{i : \boldsymbol{x}_{\mathbb{G}_i} \neq 0\}$
$\mathbb{S}_{\mathbb{G}}^c(\boldsymbol{x})$	Complement of $\mathbb{S}_{\mathbb{G}}(\boldsymbol{x})$, i.e., $\mathbb{S}_{\mathbb{G}}^c(\boldsymbol{x}) = \{i : \boldsymbol{x}_{\mathbb{G}_i} = 0\}$
$\mathbb{C}_s$	$\mathbb{C}_s = \{\boldsymbol{x} \in \mathbb{R}^n : \lVert\boldsymbol{x}\rVert_{p,0} \leq s\}$
$\lvert\mathbb{G}\rvert$	Cardinality of $\mathbb{G}$
$\lVert \cdot \rVert$	The Euclidean norm
$[\boldsymbol{x}_{\mathbb{G}}]_{p,q}$	$[\boldsymbol{x}_{\mathbb{G}}]_{p,q} = \left(\lVert\boldsymbol{x}_{\mathbb{G}_1}\rVert_p^q, \lVert\boldsymbol{x}_{\mathbb{G}_2}\rVert_p^q, \ldots, \lVert\boldsymbol{x}_{\mathbb{G}_N}\rVert_p^q\right)^T$
$[\boldsymbol{x}_{\mathbb{G}}]_{p,q}^{\downarrow}$	Elements in $[\boldsymbol{x}_{\mathbb{G}}]_{p,q}$ arranged in nonincreasing order
$([\boldsymbol{x}_{\mathbb{G}}]_{p,q}^{\downarrow})_r$	The r-th element of $[\boldsymbol{x}_{\mathbb{G}}]_{p,q}^{\downarrow}$
$\boldsymbol{x}^k$	An estimate of $\boldsymbol{x}$ after k iterations
$\langle \boldsymbol{x}, \boldsymbol{y} \rangle$	The Euclidean inner product between $\boldsymbol{x}$ and $\boldsymbol{y}$

(i) If $p = 1$, then for each element $\boldsymbol{z}_j$ in vector $\boldsymbol{z}$:

$$\text{soft}_{\frac{\lambda}{L},1}(\boldsymbol{z}_j) = \text{sign}(\boldsymbol{z}_j) \max(|\boldsymbol{z}_j| - \frac{\lambda}{L}, 0). \tag{5}$$

(ii) If $p = 2$, then

$$\text{soft}_{\frac{\lambda}{L},2}(\boldsymbol{z}) = \begin{cases} (\|\boldsymbol{z}\| - \frac{\lambda}{L}) \frac{\boldsymbol{z}}{\|\boldsymbol{z}\|} & \text{if } \|\boldsymbol{z}\| > \frac{\lambda}{L}; \\ 0 & \text{if } \|\boldsymbol{z}\| \leq \frac{\lambda}{L}. \end{cases} \tag{6}$$

C. Assumptions

Assume that f is a non-negative real-valued function, representing, for example, the difference between a model's predicted value and the actual observed value. Throughout this paper, we make the following assumptions:

(A_1) f is assumed to be continuously differentiable, and ∇f is Lipschitz continuous with Lipschitz constant L_f.

(A_2) f is bounded below.

III. Problem Reformulation and Closed-Form Solution

In this section, we first propose an equivalent weighted reformulation of the group-sparsity constraint in problem (1). We then reformulate the proposed equivalent problem into a problem with a simple constraint via the Lagrangian relaxation. By approximating the objective function with a separable quadratic function, the unconstrained problem admits closed-form solutions for some specific values of p and q, which can be used to develop iterative weighted thresholding methods.

A. Equivalent problem reformulation

Lemma 1: $\boldsymbol{x} \in \mathbb{C}_s$ is equivalent to that there exists $\boldsymbol{w} \in \{0, 1\}^N$ such that

$$\begin{cases} \langle \boldsymbol{w}, [\boldsymbol{x}_{\mathbb{G}}]_{p,q} \rangle \ \leq 0; \\ \|\mathbf{1} - \boldsymbol{w}\|_0 \leq s, \end{cases} \tag{7}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $p > 0$ and $0 < q \leq 1$.

Note that the equivalence also holds for any $q > 0$ in Lemma 1. The reason why we mainly consider the case

$0 < q \leq 1$ is that we are interested in designing a sparse induced function that can shrink small elements of a solution towards 0. In addition, we require that the first inequality in (7) be less than or equal to 0, not just equal to 0, to ensure that the Lagrange multiplier is greater than or equal to 0, which facilitates the subsequent analysis and design of our algorithm.

Obviously, the variable $\boldsymbol{w}$ in (7) can be set as

$$\begin{cases} \boldsymbol{w}_i = 0 & \text{if } i \in \mathbb{S}_{\mathbb{G}}(\boldsymbol{x}); \\ \boldsymbol{w}_i = 1 & \text{otherwise.} \end{cases}$$

In this way, no matter what strategy we use, the selected groups in $\mathbb{S}_{\mathbb{G}}(\boldsymbol{x})$ are not penalized, which helps our method to finally identify the correct support set.

Next, based on Lemma 1, we reformulate problem (1) into the following equivalent minimization problem

$$\hat{\boldsymbol{x}} = \mathop{\arg\min}_{\boldsymbol{x}\in\mathbb{R}^n, \boldsymbol{w}\in\{0,1\}^N} f(\boldsymbol{x}) \quad \text{s.t.} \begin{cases} \langle \boldsymbol{w}, [\boldsymbol{x}_{\mathbb{G}}]_{p,q} \rangle & \leq 0; \\ \|\mathbf{1} - \boldsymbol{w}\|_0 \leq s, \end{cases} \tag{8}$$

where $p > 0$, $0 < q \leq 1$.

By defining $\Omega = \{(\boldsymbol{x}, \boldsymbol{w}) : \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{w} \in \{0,1\}^N, \|\mathbf{1} - \boldsymbol{w}\|_0 \leq s\}$, we consider the Lagrangian $\mathcal{L}$ associated with problem (8):

$$\mathcal{L}_{p,q}(\boldsymbol{x}, \boldsymbol{w}, \lambda) = f(\boldsymbol{x}) + \lambda\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w}), \tag{9}$$

where $(\boldsymbol{x}, \boldsymbol{w}) \in \Omega$, $\lambda \geq 0$ is the Lagrange multiplier and

$$\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w}) = \langle \boldsymbol{w}, [\boldsymbol{x}_{\mathbb{G}}]_{p,q} \rangle = \sum_{i=1}^{N} (\boldsymbol{w}_i \times \|\boldsymbol{x}_{\mathbb{G}_i}\|_p^q). \tag{10}$$

Then we define the Lagrange dual function $g : \mathbb{R} \to \mathbb{R}$ as the minimum value of the Lagrangian $\mathcal{L}$ over $(\boldsymbol{x}, \boldsymbol{w})$:

$$g(\lambda) = \min_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \mathcal{L}_{p,q}(\boldsymbol{x}, \boldsymbol{w}, \lambda). \tag{11}$$

And the Lagrange dual problem is

$$\max_{\lambda\geq 0} g(\lambda), \tag{12}$$

which yields a lower bound on the optimal value of $f(\boldsymbol{x})$ in problem (8).

Let $(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star)$ denote an optimal solution of the problem involved in (11). That is, for a given λ, there holds

$$(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star) = \mathop{\arg\min}_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \{f(\boldsymbol{x}) + \lambda\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w})\}. \tag{13}$$

Then we analyze the strong duality property of the Lagrange dual function $g(\lambda)$.

Theorem 1: For any given λ $(\lambda \geq 0)$, let $\mathbb{C}_\lambda^\star = \{\boldsymbol{x}_\lambda^\star\}$. In the cases of $0 < p \leq 1, 0 < q \leq 1$ or $p = 2, q = 1$, suppose $\mathbb{C}_\lambda^\star$ and $\{\nabla f(\boldsymbol{x}) : \boldsymbol{x} \in \mathbb{C}_\lambda^\star\}$ are bounded, then there exists $0 < \bar{\lambda} \neq \infty$ such that $\psi_{p,q}(\boldsymbol{x}_{\bar{\lambda}}^\star, \boldsymbol{w}_{\bar{\lambda}}^\star) = 0$, and problem (1) has the strong duality property:

$$\min_{\boldsymbol{x}\in\mathbb{C}_s} f(\boldsymbol{x}) = \max_{\lambda\geq 0} g(\lambda) = \max_{\lambda\geq 0} \min_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \mathcal{L}_{p,q}(\boldsymbol{x}, \boldsymbol{w}, \lambda).$$

Benefiting from the strong duality property, we can obtain a solution of problem (1) by solving the Lagrange dual problem (12). Therefore, we first consider how to approximately solve problem (11).

B. The subproblem and its optimal solution

Applying the proximal gradient method, we can approximately solve problem (11) by iteratively solving the following sub-problem:

$$\begin{aligned} (\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) &= \mathop{\arg\min}_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \mathcal{Y}_{L_k,\lambda,p,q,\boldsymbol{x}^k}(\boldsymbol{x}, \boldsymbol{w}) \\ &= \mathop{\arg\min}_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \frac{L_k}{2}\|\boldsymbol{x} - \boldsymbol{z}^k\|^2 + \lambda\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w}), \end{aligned} \tag{14}$$

where $L_k > 0$ is an iteratively updated step-size factor and $\boldsymbol{z}^k = \boldsymbol{x}^k - \frac{1}{L_k}\nabla f(\boldsymbol{x}^k)$.

Next, we discuss how to solve the sub-problem (14) efficiently.

For each group $\mathbb{G}_i$ $(i = 1, \ldots, N)$, we consider the following low dimensional sub-problem of problem (14):

$$\min_{\boldsymbol{x}_{\mathbb{G}_i}\in\mathbb{R}^{N_i}, \boldsymbol{w}_i\in\{0,1\}} \left\{ \frac{L_k}{2}\|\boldsymbol{x}_{\mathbb{G}_i} - \boldsymbol{z}_{\mathbb{G}_i}^k\|^2 + \lambda\boldsymbol{w}_i \|\boldsymbol{x}_{\mathbb{G}_i}\|_p^q \right\}. \tag{15}$$

Obviously, the value of (p, q) can have many choices. In the rest of this paper, in the view of the simplicity of the closed-form solution as well as the widespread applications of LASSO and group LASSO, we focus on the case of $p \in \{1, 2\}$ and $q = 1$.

Lemma 2: The solution of problem (15) in the case $p \in \{1, 2\}$ and $q = 1$ is given by the following weighted group thresholding operator:

$$(T_{L_k,\lambda,p}(\boldsymbol{x}^k))_i = \begin{cases} (\boldsymbol{z}_{\mathbb{G}_i}^k, 0) & \text{if } i \in \mathbb{S}_{\mathbb{G}}^{k+1}; \\ (\text{soft}_{\frac{\lambda}{L_k},p}(\boldsymbol{z}_{\mathbb{G}_i}^k), 1) & \text{otherwise,} \end{cases} \tag{16}$$

where $i = 1, \ldots, N$, $\mathbb{S}_{\mathbb{G}}^{k+1}$ is the selected group support set at the $(k+1)$-th iteration, and $\text{soft}_{\frac{\lambda}{L_k},p}(\cdot)$ is the soft-thresholding operator defined in (5) and (6).

It can be seen from (16) that, for the selected support groups, we only perform the gradient descent step while for the other unselected groups, we subsequently perform the soft thresholding operation to shrink their magnitudes towards 0.

However, whether $T_{L_k,\lambda,p}(\boldsymbol{x}^k)$ is a solution to problem (14) depends on the choice of $\mathbb{S}_{\mathbb{G}}^{k+1}$. Therefore, a key issue in solving problem (14) is the identification of group support set. Although our weighted framework has compatibility feature for a wide range of strategies, in this work, between all C_N^s possible combinations of group choices, we evaluate the importance of different groups from the perspective of sub-problem minimization and consequently identify the group support set. More specifically, in order to obtain the optimal solution of sub-problem (14), we remark that the following two strategies, i.e., top-s groups and best-s groups, are preferable to serve the purpose.

*Definition 2 (**Top-s groups**):* Let $\boldsymbol{z} \in \mathbb{R}^n$ be a given vector and $[\boldsymbol{z}_{\mathbb{G}}]_p^\downarrow$ be the non-increasing arrangement of $\|\boldsymbol{z}_{\mathbb{G}_i}\|_p, i = 1, \ldots, N$. Top-$s$ groups strategy picks at most s-largest non-zero magnitudes of $[\boldsymbol{z}_{\mathbb{G}}]_p^\downarrow$. The detected indices form the group support set denoted by $\mathbb{A}_s(\boldsymbol{z})$, i.e.,

$$\mathbb{A}_s(\boldsymbol{z}) = \{i : ([\boldsymbol{z}_{\mathbb{G}}]_p^\downarrow)_i > 0,\ s.t.\ i \in [1, s]\}. \tag{17}$$

*Definition 3 (**Best-s groups**):* We define $\boldsymbol{y}(\boldsymbol{z}) \in \mathbb{R}^N$ as

$$(\boldsymbol{y}(\boldsymbol{z}))_i = \mathcal{Y}_{L,\lambda,p,1,\boldsymbol{z}_{\mathbb{G}_i}}(\text{soft}_{\frac{\lambda}{L},p}(\boldsymbol{z}_{\mathbb{G}_i}),1) - \mathcal{Y}_{L,\lambda,p,1,\boldsymbol{z}_{\mathbb{G}_i}}(\boldsymbol{z}_{\mathbb{G}_i},0),$$

where $p \in \{1,2\}$, $i = 1,\ldots,N$. Let $\boldsymbol{y}(\boldsymbol{z})^{\downarrow}$ be the non-increasing arrangement of $|(\boldsymbol{y}(\boldsymbol{z}))_i|, i = 1,\ldots,N$. Best-$s$ groups strategy picks at most s-largest non-zero elements of $\boldsymbol{y}(\boldsymbol{z})^{\downarrow}$. The detected indices form the group support set denoted by $\mathbb{B}_s(\boldsymbol{z})$, i.e.,

$$\mathbb{B}_s(\boldsymbol{z}) = \{i : (\boldsymbol{y}(\boldsymbol{z})^{\downarrow})_i > 0,\ s.t.\ i \in [1,s]\}. \tag{18}$$

From the above two definitions, it is clear that $|\mathbb{A}_s(\boldsymbol{z})| \leq s$ and $|\mathbb{B}_s(\boldsymbol{z})| \leq s$. Compounding the problem is that, when $|\mathbb{A}_s(\boldsymbol{z})| = s$ and $|\mathbb{B}_s(\boldsymbol{z})| = s$, $\mathbb{A}_s(\boldsymbol{z})$ and $\mathbb{B}_s(\boldsymbol{z})$ may not be unique. For problem (14), if the identified set $\mathbb{S}_{\mathbb{G}}^{k+1}$ is not unique, then by Lemma 2, $\boldsymbol{w}^{k+1}$ is not unique. Next, we show that if we set $\mathbb{S}_{\mathbb{G}}^{k+1} = \mathbb{A}_s(\boldsymbol{z}^k)$ or $\mathbb{S}_{\mathbb{G}}^{k+1} = \mathbb{B}_s(\boldsymbol{z}^k)$ in (16), then $T_{L_k,\lambda,p}(\boldsymbol{x}^k)$ is an optimal solution of problem (14) in some cases, whether $\mathbb{S}_{\mathbb{G}}^{k+1}$ is unique or not.

Theorem 2: In (16), let $\mathbb{S}_{\mathbb{G}}^{k+1} = \mathbb{A}_s(\boldsymbol{z}^k)$, then $T_{L_k,\lambda,p}(\boldsymbol{x}^k)$ is an optimal solution of problem (14) in the following cases:

(i) $p = 2, q = 1$; or

(ii) $p = 1, q = 1$ and there is only one element in each group.

Theorem 3: In (16), let $\mathbb{S}_{\mathbb{G}}^{k+1} = \mathbb{B}_s(\boldsymbol{z}^k)$, then in the case $p \in \{1,2\}, q = 1$, $T_{L_k,\lambda,p}(\boldsymbol{x}^k)$ is an optimal solution of problem (14).

Based on Theorems 2 and 3, we give the following definition.

*Definition 4 (**Optimal setting**):* We define each of the following settings as an optimal setting.

(i) $\mathbb{S}_{\mathbb{G}}^{k+1} = \mathbb{A}_s(\boldsymbol{z}^k)$ and $p = 2, q = 1$.

(ii) $\mathbb{S}_{\mathbb{G}}^{k+1} = \mathbb{A}_s(\boldsymbol{z}^k), p = 1, q = 1$, and there is only one element in each group.

(iii) $\mathbb{S}_{\mathbb{G}}^{k+1} = \mathbb{B}_s(\boldsymbol{z}^k)$ and $p \in \{1,2\}, q = 1$.

It follows from Theorems 2 and 3 that, $T_{L_k,\lambda,p}(\boldsymbol{x}^k)$ is an optimal solution to problem (14) if ($\mathbb{S}_{\mathbb{G}}^{k+1}$, p, q) matches any of the optimal settings.

IV. The IWT-GSC Framework

This section presents an iterative weighted thresholding (IWT) framework for problem (11) based on the solution of problem (14), and establishes its convergence. In addition, we analyze some important factors that affect the performance of the framework.

A. Algorithm description

Based on the solution in Section III-B, we propose an iterative weighted thresholding (IWT-GSC) algorithm (Algorithm 1) for approximately solving problem (11).

Algorithm 1: $(\boldsymbol{x}, \boldsymbol{w}) \leftarrow$ IWT-GSC($\boldsymbol{x}^0$, $\boldsymbol{w}^0$, s, λ, ϵ): Iterative Weighted Thresholding Algorithm for Group Sparsity-Constrained Optimization

```
1: choose factor $\eta > 1$, $L_{min}$ and $L_{max}$
   ($0 < L_{min} < L_{max}$), $p \in \{1,2\}$, $q \leftarrow 1$, $k \leftarrow 0$;
2: repeat
3:    choose $L_k \in [L_{min}, L_{max}]$;
4:    repeat
5:       identify $\mathbb{S}_{\mathbb{G}}^{k+1}$ under the maximum group-sparsity
         constraint $s$;
6:       $(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) \leftarrow T_{L_k,\lambda,p}(\boldsymbol{x}^k)$;
7:       $L_k \leftarrow \min\{L_{max}, \eta L_k\}$;
8:    until non-monotone line search stopping criterion
      satisfied;
9:    $k \leftarrow k + 1$;
10: until $\frac{\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|}{\|\boldsymbol{x}^k\|} < \epsilon$;
11: $(\boldsymbol{x}, \boldsymbol{w}) \leftarrow (\boldsymbol{x}^k, \boldsymbol{w}^k)$.
```

IWT-GSC is an iterative algorithm. At each iteration, IWT-GSC performs gradient descent with a selected step-size, followed by a soft thresholding operation applied only to unselected groups. In Algorithm 1, lines 4–8 use a line search method to find an appropriate step-size $1/L_k$, which controls how far the iterate moves along the gradient direction at iteration k.

Then, IWT-GSC updates the estimation $(\boldsymbol{x}^k, \boldsymbol{w}^k)$ by checking a line search condition. For each accepted solution, we use the inequality

$$\frac{\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|}{\|\boldsymbol{x}^k\|} < \epsilon$$

as a measure of accuracy, which numerically means that more iterations are not worthwhile for small improvements.

Next, we consider some factors that affect both the performance and the reliability of the algorithm, including the selection of step-size and the analysis of convergence.

B. Selection of step-size

For the proximal gradient method, the selection of step-size is an important factor in determining the sequence generated by Algorithm 1, and thus has a significant impact on the quality and efficiency of the algorithm. In Algorithm 1, a combination of the Barzilai-Borwein (BB) method [37] and a non-monotone line search strategy is used to find appropriate step-sizes, which are allowed to vary across iterations.

By letting $\Delta\boldsymbol{x}^k = \boldsymbol{x}^k - \boldsymbol{x}^{k-1}$ and $\Delta\boldsymbol{g}^k = \nabla f(\boldsymbol{x}^k) - \nabla f(\boldsymbol{x}^{k-1})$, the BB method initializes the line search step size as

$$\tau_{BB}^k = \arg\min_{\tau} \|\tau^{-1}\Delta\boldsymbol{x}^k - \Delta\boldsymbol{g}^k\| = \frac{\langle\Delta\boldsymbol{x}^k, \Delta\boldsymbol{x}^k\rangle}{\langle\Delta\boldsymbol{g}^k, \Delta\boldsymbol{x}^k\rangle}.$$

Then in line 3 of Algorithm 1, we use the following rule to initialize L_k:

$$L_k = \max\{L_{min}, \min\{L_{max}, 1/\tau_{BB}^k\}\}.$$

In line 8 of Algorithm 1, the non-monotone line search criterion is used to further validate the feasibility of the current step-size. Rather than insisting on a monotonically decreasing objective at every iteration, the non-monotone line search criterion allows for temporary increases, but ensures overall descent of the objective. To this end, similar to [36], we define c_k as a relaxation of $\mathcal{L}_{p,q}(\boldsymbol{x}^k, \boldsymbol{w}^k, \lambda)$, i.e.,

$$c_k = \frac{\sum_{j=0}^{k} \gamma^{k-j} \mathcal{L}_{p,q}(\boldsymbol{x}^j, \boldsymbol{w}^j, \lambda)}{\sum_{j=0}^{k} \gamma^{k-j}}, \tag{19}$$

where $\gamma \in (0,1)$ controls the degree of nonmonotonicity. Specifically, at the k-th iteration of Algorithm 1, nonmonotone line search selects a suitable step-size according to the following acceptance test

$$\mathcal{L}_{p,q}(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}, \lambda) \leq c_k - \frac{\varsigma}{2} L_k \|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2, \tag{20}$$

where $\varsigma \in (0,1)$ is a small positive constant.

If criterion (20) is not satisfied, then backtracking is performed, decreasing the current step-size by a factor of η (line 7 of Algorithm 1), and the inner iteration of Algorithm 1 is repeated until criterion (20) is satisfied.

C. Convergence analysis

In this subsection, we analyze the convergence property of IWT-GSC under some mild conditions.

The IWT-GSC algorithm is designed to approximately solve problem (11), which is a mixed-integer programming problem involving both continuous variables $\boldsymbol{x}$ and discrete choices $\boldsymbol{w}$. Therefore, we give the following definition, which involves not only the gradient of the objective function, but also the combinatorial nature of the binary vector $\boldsymbol{w}$.

*Definition 5 (**Partial first-order stationary point**):* $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}}) \in \Omega$ is a partial first-order stationary point of (11) with respect to $\boldsymbol{x}$ if

$$0 \in \nabla f(\hat{\boldsymbol{x}}) + \lambda \partial_{\boldsymbol{x}} \psi_{p,q}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}}).$$

Lemma 3: Assume that Assumption A_1 holds, and ($\mathbb{S}_{\mathbb{G}}^{k+1}$, p, q) matches one of the optimal settings defined in Definition 4. Given $\varsigma \in (0,1)$, the following statements hold:

(i) the non-monotone line search stopping criterion (20) is satisfied whenever $L_k \geq \frac{L_f}{1-\varsigma}$;

(ii) there exists a number $c^\star$ such that $\lim\limits_{k\to\infty} c_k = c^\star$, and the sequence $\{\boldsymbol{x}^k\}$ generated by Algorithm 1 with acceptance test (20) has $\lim\limits_{k\to\infty} \|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\| = 0$.

According to Lemma 3(i), the value of L_{max} in Algorithm 1 should satisfy $L_{max} \geq \frac{L_f}{1-\varsigma}$ to ensure that the line search can be terminated.

Under Lemma 3, we have

Theorem 4: Let $\{(\boldsymbol{x}^k, \boldsymbol{w}^k)\}$ be the sequence generated by Algorithm 1. Suppose the sequence $\{\boldsymbol{x}^k\}$ is bounded, then the following results hold:

(i) $\{\boldsymbol{x}^k\}$ is convergent.

(ii) $\{\boldsymbol{w}^k\}$ has a convergent subsequence, denoted as $\{\boldsymbol{w}^{i_k}\}$.

(iii) Let $\{\boldsymbol{x}^{i_k}\}$ be the subsequence corresponding to $\{\boldsymbol{w}^{i_k}\}$. Then any accumulation point (say $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}})$) of $\{(\boldsymbol{x}^{i_k}, \boldsymbol{w}^{i_k})\}$ is a partial first-order stationary point of (11) with respect to $\boldsymbol{x}$.

V. Homotopy Method

Section IV has presented an iterative weighted thresholding (IWT) framework for problem (11), which is the inner problem of the Lagrange dual problem (12). In this section, we first analyze some properties of the Lagrange dual function, and then resort to the homotopy algorithm based on Algorithm 1 to produce an estimation of problem (1).

A. Properties of the Lagrange dual function

By Theorem 1, there exists $0 < \bar{\lambda} \neq \infty$ such that $(\boldsymbol{x}^\star_{\bar{\lambda}}, \boldsymbol{w}^\star_{\bar{\lambda}})$ is also an optimal solution of problem (1). Unfortunately however, such a $\bar{\lambda}$ is unknown. Next, we analyze the relationship between $g(\lambda)$ and λ, which will be used for the subsequent homotopy algorithm.

Definition 6: Define $\lambda^\star$ as the smallest value in the set $\mathbb{J} = \{\lambda : \lambda \in \arg\max_{\lambda \geq 0} g(\lambda)\}$.

Theorem 5: The Lagrange dual function $g(\lambda)$ is a non-decreasing function with respect to λ.

Since $g(\lambda)$ is a non-decreasing function with respect to λ, we can reach the maximum value of $g(\lambda)$ by continuously increasing λ. One problem then is that we have to check whether the extreme point has been reached.

Theorem 6: For any $\lambda > 0$, once $\psi_{p,q}(\boldsymbol{x}^\star_\lambda, \boldsymbol{w}^\star_\lambda) = 0$, then $\lambda \in \mathbb{J} = \{\lambda : \lambda \in \arg\max_{\lambda \geq 0} g(\lambda)\}$.

Theorem 5 and Theorem 6 tell that, in order to solve problem (12), we can gradually increase λ and solve problem (11) for each λ until the condition $\psi_{p,q}(\boldsymbol{x}^\star_\lambda, \boldsymbol{w}^\star_\lambda) = 0$ is satisfied. According to the strong duality property stated in Theorem 1, the solution at this point is also the solution of problem (1).

Based on the above analysis, we next resort to the homotopy technique to find such a suitable λ.

B. HIWT-GSC algorithm

In this subsection, based on the IWT-GSC, we present the main framework of our Lagrangian method for problem (12), which is called the HIWT-GSC algorithm.

Algorithm 2: $\boldsymbol{x} \leftarrow$ HIWT-GSC($\boldsymbol{x}^0$, $\boldsymbol{w}^0$, s): Homotopy algorithm based on IWT-GSC

```
initialize $k \leftarrow 0$, select $\varepsilon$, $\lambda_0$, $\epsilon_0$, $s_0$;
choose factor $\rho > 1, \varrho < 1$;
repeat
    $(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) \leftarrow$ IWT-GSC$(\boldsymbol{x}^k, \boldsymbol{w}^k, s_k, \lambda_k, \epsilon_k)$;
    $\lambda_{k+1} \leftarrow \rho\lambda_k$, $\epsilon_{k+1} \leftarrow \varrho\epsilon_k$;
    update $s_k$ to $s_{k+1}$ until reaching the desired $s$;
    $k \leftarrow k + 1$;
until $\psi_{p,q}(\boldsymbol{x}^k, \boldsymbol{w}^k) = 0$ and $\|\nabla f(\boldsymbol{x}^k)\|_\infty / \lambda_k < \varepsilon$;
debias $\boldsymbol{x}^k$ over $\mathbb{S}^c(\boldsymbol{w}^k)$;
$\boldsymbol{x} \leftarrow \boldsymbol{x}^k$.
```

A key idea of our homotopy method is to solve problem (12) with a sequence of λ. For a fixed value of λ, we run the algorithm IWT-GSC to find an approximate solution to problem (11), and then use this solution as the starting point for IWT-GSC in the next iteration. This “warm start” strategy results in fewer iterations of IWT-GSC and gets a better solution than if we just run IWT-GSC once with a desired value of λ from a random initial solution [38].

Next, we discuss some implementation details in Algorithm 2, such as guidelines for tuning the parameters, termination criteria etc., so as to obtain satisfactory results.

1) Guidelines for parameter tuning: According to Theorem 5, Algorithm 2 searches for the optimal multiplier by starting with λ_0 and then increasing it by a constant factor $\rho > 1$, until the desired stopping conditions are satisfied. And in line 5 of Algorithm 2, we gradually increase the accuracy requirement of IWT-GSC, i.e., we decrease ϵ_k as the iteration progresses.

Moreover, if we know the exact value of s in advance, we can directly set $s_k = s$ (for all k) in line 6 of Algorithm 2. Otherwise, we give an estimate of the upper bound on s and search for an appropriate s by gradually increasing s_k.

Next, we provide an optimality analysis for the solution of Algorithm 2, and then design the halting criteria.

2) Optimality analysis: First, we introduce the definition of L-stationary point.

Definition 7: The orthogonal projection operator $P(\cdot)$ onto $\mathbb{C}_s$ is defined as

$$P_{\mathbb{C}_s}(\boldsymbol{y}) = \arg\min_{\boldsymbol{x}\in\mathbb{C}_s} \|\boldsymbol{x} - \boldsymbol{y}\|^2. \tag{21}$$

*Definition 8 (**L-stationarity** [39]):* A vector $\boldsymbol{x} \in \mathbb{C}_s$ is called an L-stationary point of problem (1) if there exists $L > 0$ such that

$$\boldsymbol{x} \in P_{\mathbb{C}_s}(\boldsymbol{x} - \frac{1}{L}\nabla f(\boldsymbol{x})). \tag{22}$$

Theorem 7: Let $(\boldsymbol{x}^\star, \boldsymbol{w}^\star)$ be an accumulation point of any convergent subsequence generated by Algorithm 1 using the top-s or best-s strategy. If $\psi_{p,1}(\boldsymbol{x}^\star, \boldsymbol{w}^\star) = 0$ ($p \in \{1, 2\}$), then the following statements hold:

(i)

$$\begin{cases} (\nabla f(\boldsymbol{x}^\star))_{\mathbb{G}_i} = 0 & \text{if } i \in \mathbb{S}^c(\boldsymbol{w}^\star); \\ |(\nabla f(\boldsymbol{x}^\star))_j| \leq \lambda & \text{if } i \in \mathbb{S}(\boldsymbol{w}^\star), \forall j \in \mathbb{G}_i. \end{cases} \tag{23}$$

(ii) $\boldsymbol{x}^\star$ is an L-stationary point of problem (1).

3) Termination criteria: By Lemma 1, if there exists $\boldsymbol{w} \in \{0, 1\}^N$ such that

$$\begin{cases} \langle \boldsymbol{w}, [\boldsymbol{x}_{\mathbb{G}}]_{p,q} \rangle & = 0; \\ \|\boldsymbol{1} - \boldsymbol{w}\|_0 \leq s, \end{cases}$$

then $\boldsymbol{x} \in \mathbb{C}_s$. Algorithm 1 identifies the group support set $\mathbb{S}_{\mathbb{G}}$ under the maximum group-sparsity constraint s, which together with operator $T_{L_k,\lambda,p}(\cdot)$ guarantee that $\|\boldsymbol{1} - \boldsymbol{w}\|_0 \leq s$. Therefore, for any solution $(\boldsymbol{x}, \boldsymbol{w})$ generated by Algorithm 1, $\langle \boldsymbol{w}, [\boldsymbol{x}_{\mathbb{G}}]_{p,q} \rangle = 0$ is a sufficient condition for $\boldsymbol{x}$ to be a feasible solution of problem (1).

Furthermore, by Theorem 7, for any accumulation point $(\boldsymbol{x}^\star, \boldsymbol{w}^\star)$ of the convergent subsequence generated by Algorithm 1, $\psi_{p,q}(\boldsymbol{x}^\star, \boldsymbol{w}^\star) = \langle \boldsymbol{w}^\star, [\boldsymbol{x}^\star_{\mathbb{G}}]_{p,q} \rangle = 0$ is a sufficient condition for $\boldsymbol{x}^\star$ to be an L-stationary point of problem (1). Therefore, we use $\langle \boldsymbol{w}^k, [\boldsymbol{x}^k_{\mathbb{G}}]_{p,q} \rangle = 0$ as a stopping criterion in Algorithm 2. In addition, by (23) in Theorem 7, we add another criterion $\|\nabla f(\boldsymbol{x}^k)\|_\infty / \lambda_k < \varepsilon$ as a measure of the accuracy.

In summary, in line 8 of Algorithm 2, we use $\psi_{p,q}(\boldsymbol{x}^k, \boldsymbol{w}^k) = 0$ and $\|\nabla f(\boldsymbol{x}^k)\|_\infty / \lambda_k < \varepsilon$ as stopping criteria to guarantee an acceptable estimation to problem (1).

4) Debiasing: In fact, our algorithm can be roughly divided into two stages. In the first stage, i.e., lines 3-8 in Algorithm 2, we use the first-order method to identify the support set. In the second stage, i.e., line 9 in Algorithm 2, we consider using more efficient method (such as the second-order method) to implement subspace minimization, which is called debiasing in Algorithm 2. It is worth mentioning that, the support set identification is an iterative process while the debiasing step can be executed only once in the algorithm.

More specifically, once an approximate solution has been obtained in the first stage, we optionally perform a debiasing step, which attempts to improve the solution quality by minimizing $f(\boldsymbol{x})$ over the chosen group support set $\mathbb{S}^c(\boldsymbol{w}^k)$. That is

$$\boldsymbol{x}^\star = \arg\min\{f(\boldsymbol{x}) : \boldsymbol{x}_{\mathbb{G}_i} = 0, \forall i \in \mathbb{S}(\boldsymbol{w}^k)\}.$$

Either a restricted newton step or a conjugate gradient procedure [38, Sec. II.I], [40, Sec. 11.3] can be used in the debiasing step.

VI. Experiments

Many formulations in machine learning, such as the least squares and logistic loss functions, typically satisfy Assumption A_1. Due to space limitations, in this section, we focus only on the widely used sparse least squares regression and sparse logistic regression problems.

We provide experimental results on simulated and real data to evaluate the accuracy, sparsity, and computational efficiency of our algorithm. All experiments were performed on a workstation with Dual Intel Xeon Processor E5-2665 (up to 3.1 GHz, 16 cores and 32 threads) and 32GB memory.

Unless otherwise specified, the experimental settings are as follows. In Algorithm 1, we set $L_{max} = \frac{1}{L_{min}} = 10^{10}$, $\eta = 2$. For non-monotone acceptance test, we set $\gamma = 0.7$, $c_0 = \mathcal{L}_{p,q}(\boldsymbol{x}^0, \boldsymbol{w}^0, \lambda)$ in (19), and $\varsigma = 10^{-20}$ in (20). In Algorithm 2, we chose zero vectors for $(\boldsymbol{x}^0, \boldsymbol{w}^0)$ and set $\varepsilon = 10^{-3}$. Unless otherwise stated, we set $s_0 = s/4$ and exponentially increased s_k by $s_{k+1} = 2s_k$ until reaching the desired value s. The number of homotopy iterations was set to 10. For simplicity, we set $\lambda_0 = 0.1, \epsilon_0 = 10^{-2}$. And the default increasing factor of the sequence $\{\lambda_k\}$ was set as $\rho = 2$ and the decreasing factor of the sequence $\{\epsilon_k\}$ was set as $\varrho = 0.2$, i.e., $\lambda_{k+1} = 2\lambda_k, \epsilon_{k+1} = 0.2\epsilon_k$. In addition to the halting criterion outlined in Algorithm 2, we also set the upper bound on the total number of iterations to 10000.

The true signal and the recovered data are denoted by $\boldsymbol{x}^\dagger$ and $\boldsymbol{x}^\star$, respectively. The tuple (m, n, N, s, σ) represents the parameters of data generation. These parameters in turn are the number of samples, the dimension of the signal, the number of groups, the number of non-zero groups and the standard deviation of the noise. When a parameter is an arithmetic progression, the notation $m_1 : \Delta m : m_2$ is used to denote the sequence of numbers starting at m_1 and ending at m_2, where each number differs from the previous one by a constant amount Δm.

We use HIWT-GSC-Bi (respectively HIWT-GSC-Ti) to denote our HIWT-GSC algorithm under the "best-s" (respectively "top-s") strategy along with $p = i$.

A. Sparse least squares regression

By setting $f(\boldsymbol{x})$ in problem (1) as the least squares function, problems such as signal recovery aim at finding an s-group-sparse solution of the underdetermined linear system. That is

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2} \|\mathbf{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 \quad s.t. \quad \|\boldsymbol{x}\|_{p,0} \leq s, \tag{24}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ is a measurement matrix and $\boldsymbol{b} \in \mathbb{R}^m$ is observations.

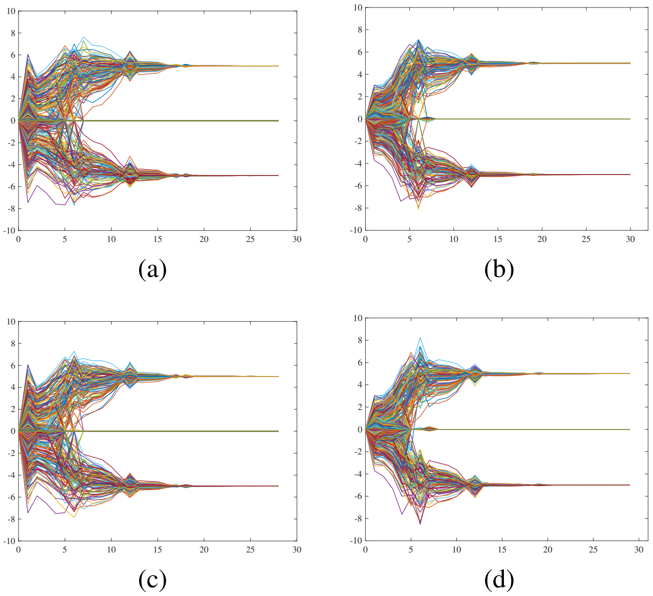
In the numerical experiments on simulated data, the data were generated as follows. First, we randomly generated an $m \times n$ matrix $\mathbf{A}$, whose entries followed an i.i.d. standard normal distribution, and each column of the matrix was normalized by the ℓ_2-norm. Then we generated a group sparse solution $\boldsymbol{x}^\dagger \in \mathbb{R}^n$ which was partitioned into N groups of equal size. We randomly picked s of the N groups as active ones, whose entries were i.i.d. random Gaussian variables, while the remaining groups were set to zero.

In the noiseless case, we generated the observation data by $\boldsymbol{b} = \mathbf{A}\boldsymbol{x}^\dagger$. In the noise case, we generated the data $\boldsymbol{b}$ by $\boldsymbol{b} = \mathbf{A}\boldsymbol{x}^\dagger + \boldsymbol{v}$, where $\boldsymbol{v}$ was an additive zero-mean Gaussian noise with standard deviation σ and was generated by the MATLAB script $\boldsymbol{v} = \sigma \times randn(m, 1)$. In this subsection, pseudo-inverse solution was used in the debiasing step and HIWT-GSC was implemented in MATLAB R2017b.

1) Convergence analysis under noiseless observation: In this experiment, the data generation setting was $(500, 2000, 500, 50, 0)$. We randomly selected the positions of the 50 non-zero groups, whose entries had the magnitudes of ± 5. We set $\lambda_0 = \|\nabla f(0)\|_\infty$ and $s_0 = s$. The number of homotopy iterations was set to 30. Fig. 1 depicts the unbiased solution trajectories that demonstrate the convergence performance of Algorithm 2. From the figure, it can be seen that as the number of iterations increases, $\boldsymbol{x}$ gradually converges to 5, -5 and 0, which are the true values of the original signal.

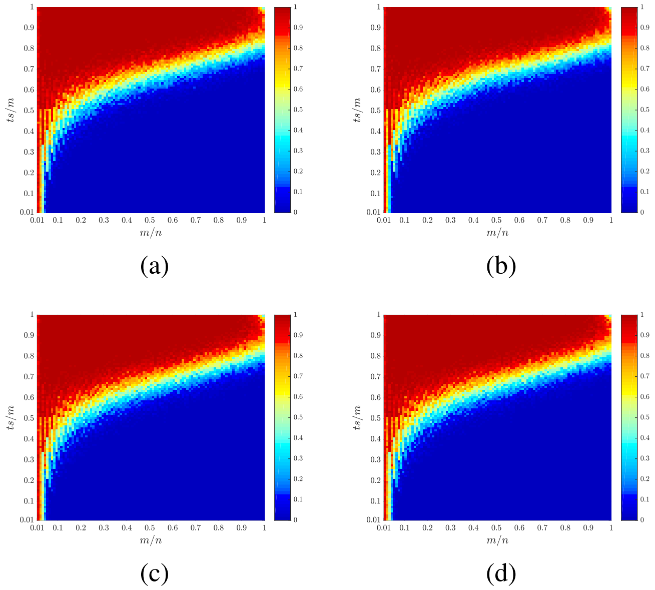
2) Performance analysis under noiseless observation: In order to more comprehensively analyze the performance of the proposed HIWT-GSC algorithm under different strategies and settings, we conducted a phase diagram study. In this experiment, we fixed the group size at $t = 4$. And for each fixed m, we varied the sparsity level from $1/N$ to $m/(Nt)$. Then, we increased m from t to n. The recovery is considered as a success when the relative error $\left\|\boldsymbol{x}^\star - \boldsymbol{x}^\dagger\right\| / \left\|\boldsymbol{x}^\dagger\right\|$ is less than 10^{-5}, and the pixel is colored with blue; otherwise, it is regarded as a failure and the pixel is colored with red. In this way, we plotted the phase diagrams in Fig. 2. As shown in Fig. 2, overall, the proposed HIWT-GSC algorithm performs well under different strategies and settings.

In addition, we calculated the differences between the successful recovery rates of different phase diagrams in Fig. 2, and presented the results in Fig. 3. As can be seen from Fig. 3a and Fig. 3b that, for more difficult examples, the successful recovery rates are higher with $p = 2$ than with $p = 1$ under the same group support set selection strategy. Meanwhile, Fig. 3c and Fig. 3d indicate that, the successful recovery rates are higher under the "best-s" strategy than under the "top-s" strategy, regardless of whether $p = 1$ or $p = 2$. Therefore, we can conclude from the experimental results that, **the HIWT-GSC algorithm is more robust with $p = 2$ under the "best-s" strategy.**



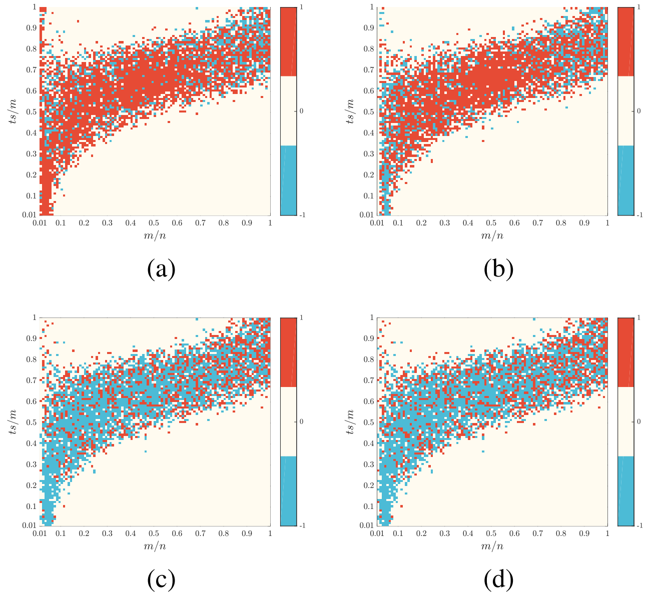
(a) (b) (c) (d)

Fig. 1. Solution trajectory of the HIWT-GSC algorithm. The horizontal axis represents the number of iterations, and the vertical axis represents the magnitude of each component in $\boldsymbol{x}$ to be restored. (a) HIWT-GSC-B1, (b) HIWT-GSC-B2, (c) HIWT-GSC-T1, (d) HIWT-GSC-T2.



(a) (b) (c) (d)

Fig. 2. Phase diagram study of the HIWT-GSC algorithm. The success cases are colored with blue and the failure cases are colored with red. The detail colors corresponding to different relative errors are marked in the color bars. The horizontal axis (m/n) represents the undersampling ratio. The vertical axis $((ts)/m)$ is the sparsity fraction. (a) HIWT-GSC-B1, (b) HIWT-GSC-B2, (c) HIWT-GSC-T1, (d) HIWT-GSC-T2.



(a) (b) (c) (d)

Fig. 3. The differences between successful recovery rates based on the phase diagram of the HIWT-GSC algorithm, which are characterized by the sign function (sign(·)) in MATLAB. The colors corresponding to the different results of the sign(·) function are marked in the color bars. The horizontal axis (m/n) represents the undersampling ratio. The vertical axis ($(ts)/m$) is the sparsity fraction. (a) sign(Fig. 2a-Fig. 2b): difference between HIWT-GSC-B1 and HIWT-GSC-B2, (b) sign(Fig. 2c-Fig. 2d): difference between HIWT-GSC-T1 and HIWT-GSC-T2, (c) sign(Fig. 2a-Fig. 2c): difference between HIWT-GSC-B1 and HIWT-GSC-T1, (d) sign(Fig. 2b-Fig. 2d): difference between HIWT-GSC-B2 and HIWT-GSC-T2.

3) Comparison of experimental results on synthetic data: This subsection presents numerical experiments on noisy observations to compare the performance of the HIWT-GSC algorithm with several state-of-the-art group sparse recovery algorithms, including homotopy iterative weighted group thresholding method [35] with $p = 1$ (HIWGT$_1$), GPDASC [32] , GOMP [31], group spectral projected gradient for L1 minimization (SPGL1)[1][41] (solves GLASSO model), SNSG [33], and GCD[2] [7] (solves group MCP model by a Group Coordinate Descent method). GSCAD [22] is not included in the comparison since the experimental results in [23] showed that GMCP outperforms GSCAD. These models or algorithms have been introduced in Section I. All downloaded codes were executed with the default settings. Given the sensitivity of SNSG to certain parameters, these parameters were set differently according to the experimental data. Our HIWT-GSC algorithm used the "best-s" strategy with $p = 2$, since it was shown in Subsection VI-A2 that it is more robust with this setting.

To evaluate the performance, we introduce three metrics, namely probability of exact support recovery (PSR), relative error, and CPU time (measured in seconds), respectively. Exact support recovery represents the ability to detect the true nonzero groups and is measured by $\mathbb{S}_{\mathbb{G}}(\boldsymbol{x}^{\star}) = \mathbb{S}_{\mathbb{G}}(\boldsymbol{x}^{\dagger})$.

Varying the group cardinality s, we generated two sets of data with data generation setting (200, 800, 200, 20:2:36, 10^{-1}) and (2000, 10000, 2500, 200:25:400, 10^{-2}). The randomly chosen non-zero groups had i.i.d. elements whose absolute values were drawn from the uniform distribution over $[1, 10]$. In this test, we set $\lambda_0 = 100$, $\mu_0 = 10, \gamma_0 = 1$ in SNSG. For a fair comparison, all algorithms in this test used the same halting condition, which is $\|\mathbf{A}\boldsymbol{x} - \boldsymbol{b}\| \leq \|\boldsymbol{v}\|$. The average results of all test algorithms over 100 replications are plotted in Fig. 4.

First, with respect to the accuracy of group support set identification, the proposed algorithm outperforms other competing algorithms as the sparsity level ($\frac{s}{N}$) increased, indicating that our algorithm has good group selection consistency. Specifically, Fig. 4a and Fig. 4b show that GPDASC and GCD achieve great group selection accuracy under relatively low sparsity levels, e.g., sampling rate $\frac{m}{n} = \frac{1}{4}$, sparsity level $\frac{s}{N} < 12\%$. However, as s increases, the PSRs of these two algorithms gradually decrease and decay more rapidly than our algorithm. Similarly, the PSR of GOMP also deteriorates significantly in a less sparse context, e.g., sampling rate=$\frac{1}{4}$, sparsity level $\frac{s}{N} > 10\%$. In comparison, the HIWGT$_1$ algorithm outperforms the GPDASC, GOMP, SNSG, and GCD algorithms in terms of PSR. Meanwhile, GSPGL1 performs poorly most of the time. These results validate that non-convex surrogates, such as those used in HIWT-GSC, HIWGT$_1$, GPDASC, and GMCP, have a superior approximation of the $\ell_{2,0}$-norm than the $\ell_{2,1}$-norm, resulting in more accurate detection of the group support set [6].

Next, in terms of the reconstruction error, as shown in Fig. 4c and Fig. 4d, the performance of the competing algorithms gradually becomes worse as their PSR decrease. It is worth noting that, although GSPGL1 performs poorly in terms of the exact recovery probability, its relative error is not too large in less sparse cases. As expected, benefiting from higher group selection accuracy, our algorithm obtains more accurate estimations in most cases.

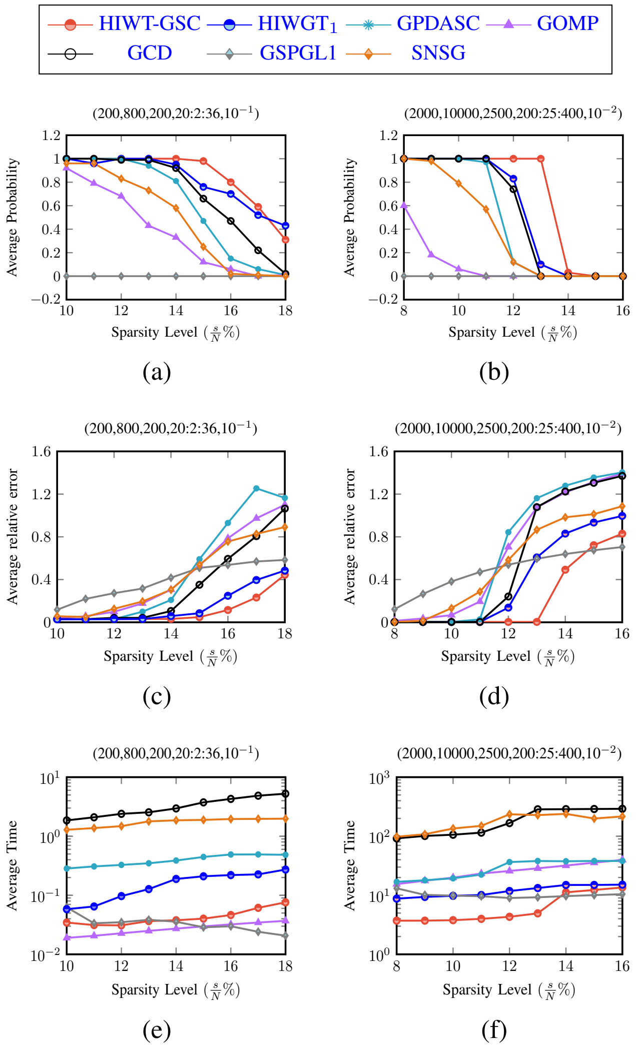
Finally, we compare computing times. Fig. 4e and Fig. 4f show that GOMP has a speed advantage on small-scale data, while this advantage gradually disappears as the size of data gets bigger. Combining convexity with continuation strategy, GSPGL1 is also computationally attractive [32], especially in less sparse cases. Due to the need to compute the subspace Newton direction, SNSG is less efficient as the data dimensions become large and the data becomes denser. Our HIWT-GSC is more speedy than GPDASC, HIWGT$_1$, SNSG, and GCD, and do have speed advantage over GOMP and GSPGL1 on large-scale data.

In summary, from these quantitative comparisons, the superiority of the proposed algorithm can be observed. Next, we apply our HIWT-GSC algorithm to some practical applications to further validate its effectiveness and robustness.

4) Application to magnetic resonance (MR) image reconstruction: The goal of this experiment is to demonstrate the effectiveness of our framework for applications involving group sparse optimization problems, such as the reconstruction of multi-contrast MR images. The experiment was carried out on multi-contrast MR images extracted from the SRI24 atlas. SRI24 is an MRI-based atlas of normal adult human brain

[1] MATLAB codes were downloaded from http://www.cs.ubc.ca/mpf/spgl1/.

[2] MATLAB codes of GPDASC, GOMP and GCD were downloaded from http://www0.cs.ucl.ac.uk/staff/b.jin/software/gpdasc.zip.



(a) (b)

(c) (d)

(e) (f)

Fig. 4. Comparison of the probability of exact support recovery (PSR), sensitivity, and running time on instances with different sparsity levels. (a), (b) Average PSR. (c), (d) Average relative error. (e), (f) Average running time.

anatomy, created using template-free population registration from high-resolution images in a group of 24 normal control subjects [42]. The tested multi-contrast MR images were acquired by multichannel-coil acquisition at a 3.0T GE scanner with three different contrast settings:

- For T1-weighted structural images: 3D axial IR-prep SPoiled Gradient Recalled (SPGR), TR = 6.5 ms, TE = 1.54 ms, slice thickness = 1.25 mm, number of slices = 124, FOV = 240×240 mm, resolution = 256×256 pixels.
- For T2-weighted and proton density-weighted images: 2D axial dual-echo fast spin echo (FSE), TR = 10,000ms, TE=14/98 ms, slice thickness = 2.5 mm, number of slices = 62, FOV = 240×240 mm, resolution = 256×256 pixels.

The MRI inverse problem can be formulated as

$$\boldsymbol{b} = \mathbf{F}\boldsymbol{x} + \boldsymbol{v} = \mathbf{F}\boldsymbol{\Phi}\boldsymbol{\alpha} + \boldsymbol{v}, \tag{25}$$

where $\boldsymbol{b}$ is the undersampled k-space data, $\boldsymbol{x}$ is the MR image to be reconstructed, $\mathbf{F}$ is the undersampled Fourier transform, $\boldsymbol{v}$ is a noise perturbation vector, $\boldsymbol{\Phi}$ is wavelet basis, and $\boldsymbol{\alpha}$ is the wavelet coefficient vector, which is assumed to be approximately s-sparse.

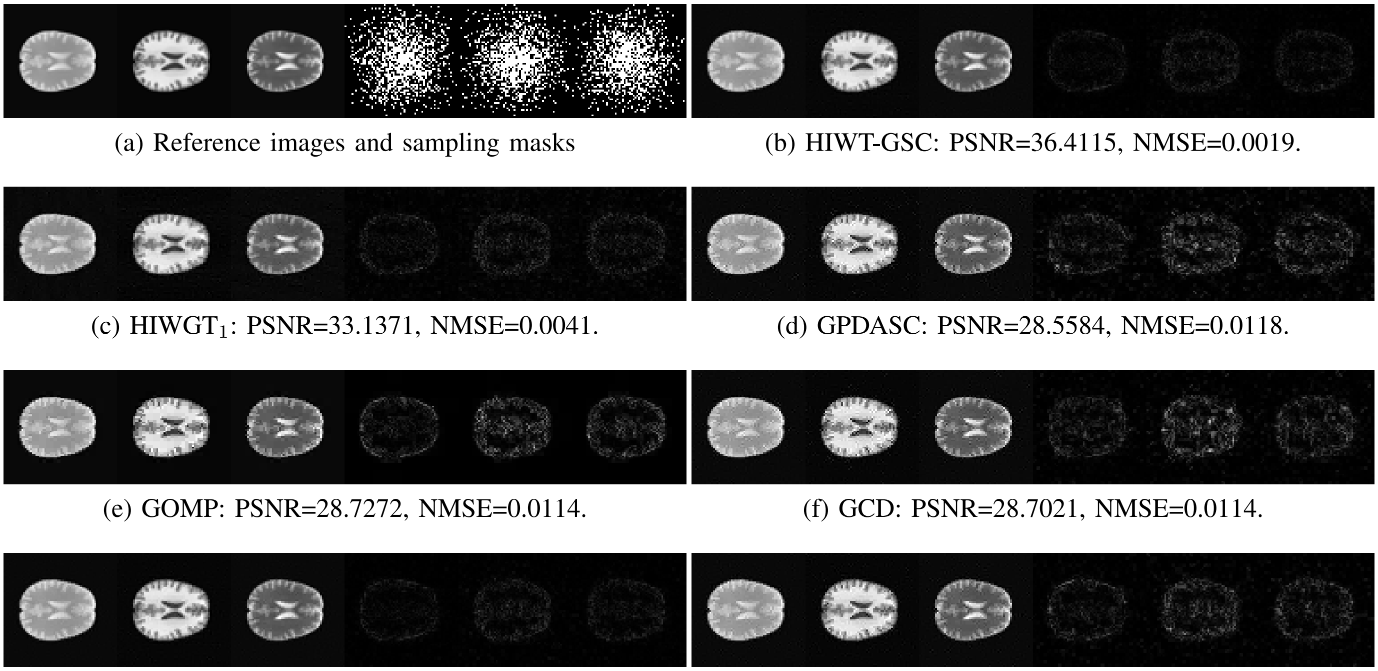
We evaluate the reconstruction accuracy in terms of the normalized mean square error (NMSE) and the Peak Signal-to-Noise Ratio (PSNR) which is measured in dB. NMSE$= \left\|\boldsymbol{x}^\star - \boldsymbol{x}^\dagger\right\|_2^2 / \left\|\boldsymbol{x}^\dagger\right\|_2^2$, where $\boldsymbol{x}^\star$ denote the reconstructed image and $\boldsymbol{x}^\dagger$ denote the reference image. PSNR $= 10 \cdot \log_{10} \frac{\mathrm{MAX}^2}{\mathrm{MSE}}$, where MAX is the maximum possible pixel value of the image. The higher the PSNR, the lower the distortion.

First, since explicit storage of the sensing matrix $\mathbf{A} = \mathbf{F}\boldsymbol{\Phi}$ (a requirement of GPDASC) is not practical at the high resolution of 256×256, we downsampled the atlas images to size 64×64 before undersampling. The selected reference images are presented in the left-hand side of Fig. 5a. Next, we undersampled the MR images using a Gaussian random sampling mask with variable density, which samples more at low frequencies and samples less at higher frequencies. The reduction factors were $R = 4$. Representative sampling masks are presented in the right-hand side of Fig. 5a. In addition, we generated $\boldsymbol{v}$ as Gaussian white noise with 0.01 standard deviation. In this way, we repeatedly performed random downsampling to test the average performance of different algorithms. **Then, utilizing the correlation among multi-contrast images of the same anatomical cross-section, we combined the same positions of the wavelet coefficients of different images into one group** [43]. Thus, we had a total of 4096 groups. Finally, we applied group sparse recovery algorithms to jointly reconstruct T1/T2-weighted MR images from their partially sampled k-space data. For fairness, in addition to the limitation of the maximum number of iterations, all group sparse recovery algorithms in this experiment used $\left\|\mathbf{A}\boldsymbol{\alpha}^k - \boldsymbol{b}\right\| \le \|\boldsymbol{v}\|$ as the stopping criterion.

By setting the upper bound of the sparsity level ($\frac{s}{N}$%) to 1%, setting the number of homotopy iterations to 3, and using "best-s" strategy with $p = 2$, we compare our method with all baseline methods. For SNSG, we set $\lambda_0 = 10000$, $\mu_0 = 1000$, $\gamma_0 = 0.5$. The results are shown in TABLE II, each of which is an average over 5 experiments. In addition, Fig. 5 depicts the representative reconstruction results. For better visualization, we highlight the reconstruction errors by magnifying them with a factor of two, i.e., $E = 2 \times |\boldsymbol{x}^\star - \boldsymbol{x}^\dagger|$, and display them in the right column of Fig. 5b-Fig. 5h.

From TABLE II, it can be seen that our algorithm outperforms the other competing ones in terms of both reconstruction accuracy and reconstruction time. In this test, $|\mathbb{S}_{\mathbb{G}}(\boldsymbol{\alpha}^\dagger)| = 1588$, which implies $\boldsymbol{\alpha}$ is not that sparse. In such a harder case, most sparse recovery algorithms fail to achieve the desired recovery quality, such as GPDASC, GOMP, GCD, and SNSG. Meanwhile, the reconstruction error of HIWGT$_1$ and GSPGL1 is relatively small in this case, which is consistent with our experimental findings in Section VI-A3. Also, as can be seen from Fig. 5, the reconstruction result of HIWGT$_1$ has more background noise compared to GSPGL1.

*5) Image classification application **with small training data** :* Solving image classification tasks with small training data remains an open challenge for modern computer vision [44]. In this subsection, we apply the proposed HIWT-GSC algorithm to the problem of object categorization with small training



(a) Reference images and sampling masks

(b) HIWT-GSC: PSNR=36.4115, NMSE=0.0019.

(c) HIWGT$_1$: PSNR=33.1371, NMSE=0.0041.

(d) GPDASC: PSNR=28.5584, NMSE=0.0118.

(e) GOMP: PSNR=28.7272, NMSE=0.0114.

(f) GCD: PSNR=28.7021, NMSE=0.0114.

(g) GSPGL1: PSNR=35.7022, NMSE=0.0023.

(h) SNSG: PSNR=30.5622, NMSE=0.0075.

Fig. 5. Representative reconstruction results for multi-contrast MR images from SRI24 atlas. (a) Atlas images at Nyquist rate sampling (left) and the sampling masks (right). (b), (c), (d), (e), (f), (g), (h) The reconstruction results (left) and the magnified absolute errors (right) of the proposed HIWT-GSC algorithm, HIWGT$_1$ [35], GPDASC [32], GOMP [31], GCD [7], GSPGL1 [41], and SNSG [33] respectively.

TABLE II
AVERAGE RECONSTRUCTION PERFORMANCE FOR A SET OF MULTI-CONTRAST MR IMAGES FROM SRI24 ATLAS BY DIFFERENT METHODS. N=4096, $|\mathbb{S}_{\mathbb{G}}(\boldsymbol{\alpha}^{\dagger})| = 1588$, $\boldsymbol{\upsilon} = 1e-2$.

Algorithm	Time (sec.)	PNSR	NMSE
HIWT-GSC	**26.24**	**36.60**	**0.0019**
HIWGT$_1$ [35]	47.46	33.31	0.0040
GPDASC [32]	200.24	28.91	0.0111
GOMP [31]	102.21	28.44	0.0122
GCD [7]	922.98	28.99	0.0107
GSPGL1 [41]	51.12	35.81	0.0022
SNSG [33]	310.49	30.79	0.0071

data and design a new support set identification strategy for the application. Several image databases, including the Olivetti Research Laboratory (ORL) database [45], the Extended Yale B (ExYaleB) database [46] and the Columbia Object Image Library (COIL20) database [47], are selected for evaluation.

The ORL database[3] contains a set of face images from 40 distinct subjects, each with ten different images. For some subjects, the images were taken at different times, varying the lighting, facial expressions, and facial details. All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position. The size of each image is 92 × 112 pixels, with 256 gray levels per pixel. In our experiment, each image was downsampled to a size of 46×56 pixels.

The ExYaleB database[4] contains 2414 frontal-face images of size 168×192 from 38 human subjects. There are approximately 64 images for each subject. The images were taken under different illuminations and various facial expressions. In our experiment, each image was resized to 32×32.

The COIL-20 database[5] contains 1,440 normalized grayscale images of 20 objects. Each object has 72 images which were taken at pose intervals of 5 degrees in a 360 rotation with a size of 32×32.

We employed the sparse representation based classification (SRC) [48] method to construct dictionaries whose base elements (features) are the training samples themselves. The SRC method assumes that a test sample can be sufficiently represented by those training samples from the same subject [48]. **Therefore, it is natural to stack the training samples of a subject by grouping them according to their classes.**

Obviously, a critical issue in this experiment is the dimension of the feature space. First, similar to the experiment in [49], the principle component analysis (PCA) algorithm was used as a preprocessing step to extract face features. Then, we measured the robustness of the different methods by varying the dimension of the extracted features in the dictionary, which was set to be one less than the number of training samples. For each fixed number of training samples, we applied sparse optimization algorithms to explore the coefficients that represent the test sample as a linear combination of training samples.

[3] ORL database can be downloaded from https://cam-orl.co.uk/facedatabase.html.

[4] ExYaleB database can be downloaded from https://www.kaggle.com/datasets/tbourton/extyalebcroppedpng.

[5] COIL-20 database can be downloaded from https://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php.

After that, we calculated the reconstruction residuals of each class according to the representation coefficients. Finally, a test sample was classified into the object class that minimizes the reconstruction residual.

As mentioned above, our weighted model is compatible with a diversity of strategies so as to better fit in different applications. With this in mind, we adopt a hybrid strategy to identify the group support set in this test. Specifically, in line 5 of Algorithm 1, we set

$$\mathbb{S} = \mathbb{B}_s(\boldsymbol{z}) \cup \bar{\mathbb{A}}_s(\boldsymbol{z})$$

where $\bar{\mathbb{A}}_s(\boldsymbol{z}) = \{i : (|\boldsymbol{z}|^{\downarrow})_j > 0,\ s.t.\ j \in [1, s], j \in \mathbb{G}_i\}$ select the groups with the s-largest non-zero individual coefficients. Besides, the other parameters in this test were set as follows. $p = 2$. For the ORL database and COIL20 database, $(\frac{s}{N}\%)$ was set to 10% while for the ExYaleB database, $(\frac{s}{N}\%)$ was set to 60%. $s_0 = \Delta s = \frac{s}{10}$, and s_k was linearly increased by $s_{k+1} = s_k + \Delta s$. For SNSG, we set $\lambda_0 = 1$, $\mu_0 = 1$, $\gamma_0 = 0.1$. The stopping tolerance ϵ in Algorithm 1 was set as $\epsilon = 1e-2$. All group sparse recovery algorithms used $\left\|\mathbf{A}\boldsymbol{x}^k - \boldsymbol{b}\right\| \leq \upsilon$ as a stopping criterion. Considering the trade-off between speed and accuracy, υ was set to 0.2.

For all databases, we randomly selected several samples per category as training samples and the rest as testing samples. And for each fixed number of training samples, the experiments were repeated 10 times with random splitting of the datasets. The average results are summarized in Fig. 6.

As can be seen from Fig. 6, there does not exist one extraordinary algorithm that can achieve the best classification accuracy on all databases, which is consistent with the conclusion in [49]. For example, the GOMP and GCD algorithms, which achieve better classification accuracy on the ExYaleB database, do not perform well on the COIL20 database. On the other hand, SNSG considers both element and group sparsity together and performs well on the COIL20 database but poorly on the ExYaleB database. Although the HIWGT$_1$ algorithm performs well on simulated data, in practice it is not so easy to obtain optimal results unless the value of the hyperparameter ε is adjusted for each data set. Thanks to the flexibility to customize the support set identification strategy, our HIWT-GSC algorithm generally performs well in most cases, especially on the COIL20 database.

B. Sparse logistic regression

Logistic regression (LR) is a popular classifier for many applications [50]. Suppose we have a training data set of length m, i.e., $\mathbb{D} = \{(\boldsymbol{a}_1, y_1), (\boldsymbol{a}_2, y_2), \ldots, (\boldsymbol{a}_m, y_m)\}$, where $\boldsymbol{a}_i = (a_{i1}, a_{i2}, \ldots, a_{in})$ is the i-th input pattern containing n features and y_i is the corresponding label taking the value of 0 or 1. Then the sigmoid function calculates the probability that $\boldsymbol{a}_i$ belongs to the class y_i by

$$p(y_i|\boldsymbol{a}_i, \boldsymbol{x}, c) = \frac{\exp(y_i(\langle \boldsymbol{a}_i, \boldsymbol{x}\rangle + c))}{1 + \exp(\langle \boldsymbol{a}_i, \boldsymbol{x}\rangle + c)}, \tag{26}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the model parameter to be learned and $c \in \mathbb{R}$ is an intercept.

Replacing $f(\boldsymbol{x})$ in (1) with a log-likelihood function, the group sparse logistic regression (LR) problem is defined as

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} L(\boldsymbol{x}) \quad s.t. \quad \|\boldsymbol{x}\|_{p,0} \leq s, \tag{27}$$

where $L(\boldsymbol{x})$ is the average logistic loss function which is defined as

$$L(\boldsymbol{x}) = -\frac{1}{m}\sum_{i=1}^{m} \log p(y_i|\boldsymbol{a}_i, \boldsymbol{x}, c).$$

We next conduct experiments to show the performance of our HIWT-GSC algorithm in solving problem (27). In this part, HIWT-GSC has been mixed-programmed by R and C languages using R version 3.5.3. Using the "best-s" strategy with $p = 2$, we compared our implementation (without debiasing step) with a publicly available R package **grpreg**[6] [23]. In this package, group selection methods for logistic regression, such as group LASSO (GLASSO), group MCP (GMCP), and group SCAD (GSCAD), are solved using group descent algorithms. All the codes in the **grpreg** package were run with default settings and their regularization parameters λ were selected along the regularization path of a fitted object according to the BIC criteria.

1) Performance on Synthetic Data: Using simulated data, we evaluate the performance of these methods in terms of group selection accuracy. As a measure we introduce the F_1-score, which is computed as follows with respect to Precision (P) and Recall (R):

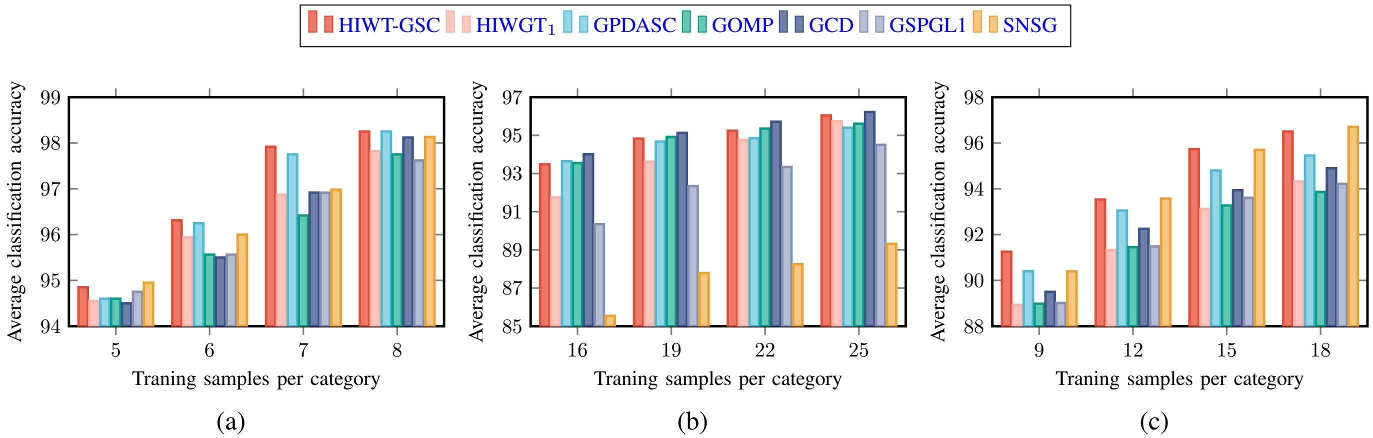
$$F_1 = \frac{2PR}{P+R}, P = \frac{|\mathbb{S}_G(\boldsymbol{x}^\star) \cap \mathbb{S}_G(\boldsymbol{x}^\dagger)|}{|\mathbb{S}_G(\boldsymbol{x}^\star)|}, R = \frac{|\mathbb{S}_G(\boldsymbol{x}^\star) \cap \mathbb{S}_G(\boldsymbol{x}^\dagger)|}{|\mathbb{S}_G(\boldsymbol{x}^\dagger)|}.$$

A sample size of $m = 5000$ was fixed throughout this test, while the number of features and groups were varied. We generated two sets of data with data generation settings (5000, 1000, 200, 10:2:30, 0) and (5000, 10000, 1000, 50:10:150, 0). Each element in the randomly picked non-zero groups was drawn independently in the standard Gaussian distribution. In addition, independent of the sparse parameter $\boldsymbol{x}$, an intercept $c \in \mathbb{R}$ was also taken into account, which was generated following the distribution $\mathcal{N}(0, 0.5^2)$. And each data sample was an independent instance of random vector $\boldsymbol{a}_i \in \mathcal{N}(0, 1)$. Then the corresponding label $\boldsymbol{y} \in \{0, 1\}^m$ was randomly generated according to the Bernoulli distribution as defined in Equation (26).

Fig. 7 reports the average results of 100 simulation runs. We can observe that as we increase s, our method achieves higher identification accuracy of active groups, which means our method correctly identifies significant groups and features.

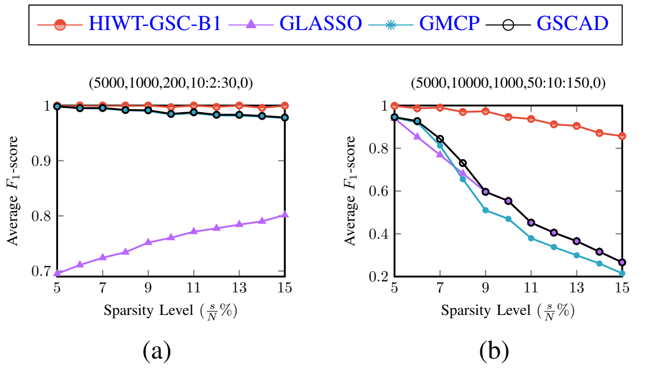
2) Performance on Real Data: To further evaluate the performance of our method on the tasks of feature selection as well as classifier design, we report experimental results on real datasets. We evaluate the performance of all the tested algorithms in terms of the $L(\boldsymbol{x})$, training time (measured in seconds), the number of selected groups (♯Groups) and the

[6] grpreg-3.3.1 codes can be downloaded from https://github.com/pbreheny/grpreg.



(a) (b) (c)

Fig. 6. Comparison of average classification accuracy with different numbers of training samples in image classification experiment. By customizing the new support set identification strategy, the HIWT-GSC algorithm performs well in classification tasks on tested datasets. (a) ORL database, (b) ExYaleB database, (c) COIL-20 database.



(a) (b)

Fig. 7. Comparison of group selection accuracy on synthetic data with different sparsity levels.

error rate (ER) on out-of-sample test set. The ER is calculated by

$$ER = \sum_{i=1}^{m} \|sign(\langle \boldsymbol{a}_i, \boldsymbol{x} \rangle + c) - y_i\|_0 \,/m \times 100\%,$$

where $c \in \mathbb{R}$ is an intercept and $sign(\cdot)$ is the sign function defined as

$$sign(t) = \begin{cases} 1 & \text{if } t > 0; \\ 0 & \text{otherwise.} \end{cases}$$

This experiment considers the problem of splice site detection, which plays an important role in gene finding algorithms. Splice sites are the regions between introns (non-coding region) and exons (coding region) DNA segments. The $5'$ end and the $3'$ end of an intron are called the donor splice site and the acceptor splice site, respectively. The canonical donor splice sites are characterized by the presence of ‘GT’ at the first two intron positions, whereas the canonical acceptor site have ‘AG’ present at the end of the intron.

The MEMset Donor dataset[7] [51] has a training set containing 8415 true (encoded as y=1) and 179438 false (encoded as y=0) human donor sites, and has an additional testing set containing 4208 true and 89717 false donor sites. The original MEMset dataset was used to build a smaller balanced training set with 6396 true and 6404 false donor sites, and a smaller testing set with 1604 true and 1596 false donor sites. In our case, the expression levels of 939 genes were recorded, which were divided into 64 groups.

Our goal is to choose the right set of genes to design a classifier that correctly classifies unseen examples, i.e., true from decoy splice sites. In this experiment, for a fair comparison, we set $s = 24$, which is the minimum number of feature groups selected by the competing algorithms. The results of the respective algorithms on the MEMset are reported in TABLE III.

TABLE III
COMPARISON OF PERFORMANCE ON MEMSET DATA SET

Algorithm	HIWT-GSC	GLASSO	GMCP	GSCAD
ER	1.08e-1	**9.82e-2**	1.10e-1	1.11e-1
$L(\boldsymbol{x})$	2.38e-1	**2.14e-1**	2.42e-1	2.43e-1
Training Time (sec.)	**5.21**	43.49	63.49	72.10
♯ Groups	**24**	51	**24**	27

We can observe that the group LASSO produced the largest models with about 80% selected features while the other three algorithms produced more sparse models with about 40% selected features. Although a smaller model may lead to a larger prediction error, it makes the prediction model more interpretable and less costly to use. Therefore, considering the advantage of producing models of reasonable size, prediction performance measured in terms of the out-of-sample error rate was best for HIWT-GSC, followed by the group MCP and the group SCAD. Moreover, HIWT-GSC achieved lower logistic loss on the training set and required less training time than the group MCP and the group SCAD. These results provide additional evidence for the effectiveness of the proposed algorithm in solving feature selection problems.

VII. Conclusions

In this work, we have proposed and analyzed a general and effective framework for solving the group-sparsity constrained

[7] MEMset Donor dataset can be downloaded from http://hollywood.mit.edu/burgelab/maxent/ssdata/.

optimization problem, which not only ensures accurate solutions but also allows for a wide range of support set identification strategies. By reformulating the group-sparsity constrained optimization problem as an equivalent mixed-integer programming problem, we introduced a Lagrange dual framework for the reformulated problem, and proposed an efficient weighted group thresholding algorithm with homotopy. Theoretically, we have established the convergence of the proposed IWT-GSC algorithm under some mild conditions. Meanwhile, we have provided a guarantee that the solution of our HIWT-GSC algorithm is an L-stationary point of the original problem.

Comprehensive numerical simulations have been performed to evaluate the performance of our algorithm. In addition, extensive experiments on publicly available data sets have been conducted in comparison with several state-of-the-art group sparse optimization approaches. The experimental results validate the effectiveness and efficiency of our algorithm and indicate that our approach has great potential in broad data processing.

Finally, it would be interesting in future work to evaluate our method on more applications. For example, we can introduce our weighted group sparse regularization term into deep neural networks (DNN) to learn a compact structure. We can also combine our method with some other image processing methods, such as the total variation (TV) method, and apply the composite model to MR image reconstruction.

Acknowledgment

We would like to express our sincere gratitude to the esteemed editors and the dedicated anonymous reviewers for their insightful suggestions and constructive feedback. Their contributions have been instrumental in enhancing the rigor and clarity of this manuscript.

References

[1] M. Elad, M. A. T. Figueiredo, and Y. Ma, “On the role of sparse and redundant representations in image processing,” *Proc. IEEE*, vol. 98, no. 6, pp. 972–982, Jun. 2010.

[2] M. Scetbon, M. Elad, and P. Milanfar, “Deep k-SVD denoising,” *IEEE Trans. Image Process.*, vol. 30, pp. 5944–5955, 2021.

[3] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, “Sparse representation for computer vision and pattern recognition,” *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.

[4] M. Yamac, M. Ahishali, A. Degerli, S. Kiranyaz, M. E. H. Chowdhury, and M. Gabbouj, “Convolutional sparse support estimator-based COVID-19 recognition from x-ray images,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 1810–1820, May 2021.

[5] M.-G. Gong, J. Liu, H. Li, Q. Cai, and L.-Z. Su, “A multiobjective sparse feature learning model for deep neural networks,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 12, pp. 3263–3277, Dec. 2015.

[6] X. Zhang, J. Zheng, D. Wang, G. Tang, Z. Zhou, and Z. Lin, “Structured sparsity optimization with non-convex surrogates of $\ell_{2,0}$-norm: A unified algorithmic framework,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–18, 2022.

[7] J. Huang, P. Breheny, and S. Ma, “A selective review of group selection in high-dimensional models,” *Stat. Sci.*, vol. 27, no. 4, pp. 481–499, Nov. 2012.

[8] M. Cheng, C. Wang, and J. Li, “Single-image super-resolution in RGB space via group sparse representation,” *IET Image Process.*, vol. 9, no. 6, pp. 461–467, Jun. 2015.

[9] S. Scardapane, D. Comminiello, A. Hussain, and A. Uncini, “Group sparse regularization for deep neural networks,” *Neurocomputing*, vol. 241, pp. 81–89, 2017.

[10] T. Chen, B. Ji, T. Ding, *et al.*, “Only train once: A one-shot neural network training and pruning framework,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 19 637–19 651.

[11] Y. Nardi and A. Rinaldo, “On the asymptotic properties of the group lasso estimator for linear models,” *Electron. J. Stat.*, vol. 2, pp. 605–633, 2008.

[12] R. Jenatton, J.-Y. Audibert, and F. Bach, “Structured variable selection with sparsity-inducing norms,” *J. Mach. Learn. Res.*, vol. 12, pp. 2777–2824, 2011.

[13] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, “Structured sparsity through convex optimization,” *Stat. Sci.*, vol. 27, no. 4, pp. 450–468, Nov. 2012.

[14] M. Stojnic, F. Parvaresh, and B. Hassibi, “On the reconstruction of block-sparse signals with an optimal number of measurements,” *IEEE Trans. Signal Process.*, vol. 57, no. 8, pp. 3075–3085, 2009.

[15] 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, 2009.

[16] X. Xu and M. Ghosh, “Bayesian variable selection and estimation for group lasso,” *Bayesian Analysis*, vol. 10, no. 4, Dec. 1, 2015.

[17] A. Beck and N. Hallak, “Optimization problems involving group sparsity terms,” *Math. Program.*, vol. 178, no. 1, pp. 39–67, Nov. 2019.

[18] F. E. Curtis, Y. Dai, and D. P. Robinson, “A subspace acceleration method for minimization involving a group sparsity-inducing regularizer,” *SIAM J. Optim.*, vol. 32, no. 2, pp. 545–572, Jun. 2022.

[19] Z. Zha, B. Wen, X. Yuan, J. Zhou, and A. C. Kot, “Low-rankness guided group sparse representation for image restoration,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 10, pp. 7593–7607, 2023.

[20] R. Wang, J. Bian, F. Nie, and X. Li, “Nonlinear feature selection neural network via structured sparse regularization,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 11, pp. 9493–9505, 2023.

[21] M. Yuan and Y. Lin, “Model selection and estimation in regression with grouped variables,” *J. R. Stat. Soc. Ser. B-Stat. Methodol.*, vol. 68, no. 1, pp. 49–67, Feb. 2006.

[22] L.-F. Wang, H.-Z. Li, and J.-H. Z. Huang, “Variable selection in nonparametric varying-coefficient models for analysis of repeated measurements,” *J. Am. Stat. Assoc.*, vol. 103, no. 484, pp. 1556–1569, Dec. 2008.

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

[24] Y. Wang and W. Yin, “Sparse signal reconstruction via iterative support detection,” *SIAM J. Imaging Sci.*, vol. 3, no. 3, pp. 462–491, Jan. 2010.

[25] Y. Hu, D.-B. Zhang, J.-P. Ye, X.-L. Li, and X.-F. He, “Fast and accurate matrix completion via truncated nuclear norm regularization,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 9, pp. 2117–2130, Sep. 2013.

[26] Z. Lu and X. Li, “Sparse recovery via partial regularization: Models, theory, and algorithms,” *Math. Oper. Res.*, vol. 43, no. 4, pp. 1290–1316, Nov. 2018.

[27] Q. Feng, J. Wang, and F. Zhang, “Block-sparse signal recovery based on truncated ℓ_1 minimisation in non-gaussian noise,” *IET Commun.*, vol. 13, no. 2, pp. 251–258, Jan. 2019.

[28] L. Pan and X. Chen, “Group sparse optimization for images recovery using capped folded concave functions,” *SIAM J. Imaging Sci.*, vol. 14, no. 1, pp. 1–25, Jan. 2021.

[29] X. Zhang and D. Peng, “Solving constrained nonsmooth group sparse optimization via group capped-ℓ_1 relaxation and group smoothing proximal gradient algorithm,” *Comput. Optim. Appl.*, vol. 83, no. 3, pp. 801–844, Dec. 2022.

[30] Y. Cao, L. Kang, X. Li, Y. Liu, Y. Luo, and Y. Shi, “Newton-raphson meets sparsity: Sparse learning via a novel penalty and a fast solver,” *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Mar. 9, 2023, doi:10.1109/TNNLS.2023.3251748.

[31] Z. Ben-Haim and Y. C. Eldar, “Near-oracle performance of greedy block-sparse estimation techniques from noisy measurements,” *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 5, pp. 1032–1047, Sep. 2011.

[32] Y.-L. Jiao, B.-T. Jin, and X.-L. Lu, “Group sparse recovery via the $\ell^0(\ell^2)$ penalty: Theory and algorithm,” *IEEE Trans. Signal Process.*, vol. 65, no. 4, pp. 998–1012, Feb. 2017.

[33] S. Liao, C. Han, T. Guo, and B. Li, “Subspace newton method for sparse group ℓ_0 optimization problem,” *J. Global Optim.*, early access, Apr. 29, 2024, doi:10.1007/s10898-024-01396-y.

[34] W.-X. Zhu, H.-T. Huang, L.-F. Jiang, and J.-L. Chen, "Weighted thresholding homotopy method for sparsity constrained optimization," *J. Comb. Optim.*, vol. 44, no. 3, pp. 1924–1952, 2022.

[35] L.-F. Jiang and W.-X. Zhu, "Iterative weighted group thresholding method for group sparse recovery," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 63–76, 2021.

[36] H. Li and Z.-C. Lin, "Accelerated proximal gradient methods for nonconvex programming," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 379–387.

[37] J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," *IMA J. Numer. Anal.*, vol. 8, no. 1, pp. 141–148, Jan. 1988.

[38] S. Wright, R. Nowak, and M. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2479–2493, Jul. 2009.

[39] A. Beck and Y. C. Eldar, "Sparsity constrained nonlinear optimization: Optimality conditions and algorithms," *SIAM J. Optim.*, vol. 23, no. 3, pp. 1480–1509, Jan. 2013.

[40] G. H. Golub and C. F. Van Loan, *Matrix computations*, Fourth edition. Baltimore: The Johns Hopkins University Press, 2013, 756 pp.

[41] E. van den Berg and M. P. Friedlander, "Probing the pareto frontier for basis pursuit solutions," *SIAM J. Sci. Comput.*, vol. 31, no. 2, pp. 890–912, Jan. 2009.

[42] T. Rohlfing, N. M. Zahr, E. V. Sullivan, and A. Pfefferbaum, "The SRI24 multichannel atlas of normal adult human brain structure," *Human Brain Mapping*, vol. 31, no. 5, pp. 798–819, May 2010.

[43] J. Huang, C. Chen, and L. Axel, "Fast multi-contrast MRI reconstruction," *Magn. Reson. Imaging*, vol. 32, no. 10, pp. 1344–1352, Dec. 2014.

[44] L. Brigato and S. Mougiakakou, "No data augmentation? alternative regularizations for effective training on small datasets," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2023, pp. 139–148.

[45] F. Samaria and A. Harter, "Parameterisation of a stochastic model for human face identification," in *Proc. 2nd IEEE Workshop Appl. Comput. Vis.*, 1994, pp. 138–142.

[46] A. Georghiades, P. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.

[47] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (coil-20)," Dept. Comput. Sci., Columbia Univ., Tech. Rep. CUCS-005-96, Feb. 1996.

[48] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Yi Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[49] Z. Zhang, Y. Xu, J. Yang, X.-L. Li, and D. Zhang, "A survey of sparse representation: Algorithms and applications," *IEEE Access*, vol. 3, pp. 490–530, May 2015.

[50] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied logistic regression* (Wiley series in probability and statistics), third edition. Hoboken, NJ: Wiley, 2013, pp. 227–376.

[51] G. Yeo and C. B. Burge, "Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals," *J. Comput. Biol.*, vol. 11, no. 2, pp. 377–394, Mar. 2004.

APPENDICES

In this supplementary file, we present the proofs of lemmas and theorems in the paper.

CONTENTS

Appendix A: Proof of Lemma 1 1

Appendix B: Proof of Theorem 1 1

Appendix C: Proof of Lemma 2 2

Appendix D: Proof of Theorem 2 2

Appendix E: Proof of Theorem 3 3

Appendix F: Proof of Lemma 3 4

Appendix G: Proof of Theorem 4 5

Appendix H: Proof of Theorem 5 5

Appendix I: Proof of Theorem 6 5

Appendix J: Proof of Theorem 7 6

Appendix K: Empirical study on the number of inner and outer iterations of the algorithm 6

Appendix L: Application to electroencephalography signal reconstruction 7

APPENDIX A
PROOF OF LEMMA 1

Proof: We set the value of each element in $\boldsymbol{w}$ as

$$\begin{cases} \boldsymbol{w}_i = 0 & \text{if } i \in \mathbb{S}_{\mathbb{G}}(\boldsymbol{x}); \\ \boldsymbol{w}_i = 1 & \text{otherwise.} \end{cases} \tag{28}$$

Combinding (28) with $\boldsymbol{x} \in \mathbb{C}_s$, we have that there exists $\boldsymbol{w} \in \{0,1\}^N$ such that $< \boldsymbol{w}, [\boldsymbol{x}_{\mathbb{G}}]_{p,q} >= 0$ and $\|\mathbf{1} - \boldsymbol{w}\|_0 \leq s$. Hence Equation (7) holds.

Conversely, if there exists $\boldsymbol{w} \in \{0,1\}^N$ such that $\|\mathbf{1} - \boldsymbol{w}\|_0 \leq s$, then the number of zero components in $\boldsymbol{w}$ is no more than s. By $< \boldsymbol{w}, [\boldsymbol{x}_{\mathbb{G}}]_{p,q} >\leq 0$, we have $\boldsymbol{w}_i = 0$ if $i \in \mathbb{S}_{\mathbb{G}}(\boldsymbol{x})$. These indicate that the number of nonzero groups in $\boldsymbol{x}$ is no more than s, i.e., $\boldsymbol{x} \in \mathbb{C}_s$. ■

APPENDIX B
PROOF OF THEOREM 1

First, we introduce some lemmas.

Lemma 4: Let $(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star)$ be an optimal solution of $g(\lambda)$, then $(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star)$ satisfies the following expression

(i)

$$(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star) = \operatorname*{arg\,min}_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \Big\{ \frac{L}{2}\|\boldsymbol{x} - \boldsymbol{z}_\lambda^\star\|^2 + \lambda\psi_{p,q}(\boldsymbol{x},\boldsymbol{w}) \Big\}, \tag{29}$$

where $\boldsymbol{z}_\lambda^\star = \boldsymbol{x}_\lambda^\star - \frac{1}{L}\nabla f(\boldsymbol{x}_\lambda^\star)$, $L \geq L_f$.

(ii) In the case of $p \in \{1,2\}$ and $q = 1$,

$$((\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}, (\boldsymbol{w}_\lambda^\star)_i) = \begin{cases} ((\boldsymbol{z}_\lambda^\star)_{\mathbb{G}_i}, 0) & \text{if } i \in \mathbb{S}^c(\boldsymbol{w}_\lambda^\star); \\ (\text{soft}_{\frac{\lambda}{L},p}((\boldsymbol{z}_\lambda^\star)_{\mathbb{G}_i}), 1) & \text{otherwise,} \end{cases} \tag{30}$$

where the $\text{soft}_{\frac{\lambda}{L},p}(\cdot)$ is the soft-thresholding operators defined in (5) and (6).

Proof: (i) According to the definition of $(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star)$ in (13), for any feasible solution $(\boldsymbol{x}, \boldsymbol{w})$ of problem (9), we have

$$f(\boldsymbol{x}_\lambda^\star) + \lambda\psi_{p,q}(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star) \leq f(\boldsymbol{x}) + \lambda\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w}).$$

Since $\nabla f(\boldsymbol{x})$ is Lipschitz continuous, for $L \geq L_f$ it holds that

$$f(\boldsymbol{x}) \leq f(\boldsymbol{x}_\lambda^\star) + \langle \nabla f(\boldsymbol{x}_\lambda^\star), \boldsymbol{x} - \boldsymbol{x}_\lambda^\star \rangle + \frac{L}{2}\|\boldsymbol{x} - \boldsymbol{x}_\lambda^\star\|^2.$$

The above two inequalities imply that

$$\begin{aligned} &\langle \nabla f(\boldsymbol{x}_\lambda^\star), \boldsymbol{x} - \boldsymbol{x}_\lambda^\star \rangle + \frac{L}{2}\|\boldsymbol{x} - \boldsymbol{x}_\lambda^\star\|^2 + \lambda\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w}) \\ \geq& \langle \nabla f(\boldsymbol{x}_\lambda^\star), \boldsymbol{x}_\lambda^\star - \boldsymbol{x}_\lambda^\star \rangle + \frac{L}{2}\|\boldsymbol{x}_\lambda^\star - \boldsymbol{x}_\lambda^\star\|^2 + \lambda\psi_{p,q}(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star), \end{aligned}$$

which means

$$\begin{aligned} (\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star) = \operatorname*{arg\,min}_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \Big\{ &\langle f(\boldsymbol{x}_\lambda^\star), \boldsymbol{x} - \boldsymbol{x}_\lambda^\star \rangle + \\ &\frac{L}{2}\|\boldsymbol{x} - \boldsymbol{x}_\lambda^\star\|^2 + \lambda\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w}) \Big\}. \end{aligned}$$

By noting that

$$\begin{aligned} &\|\boldsymbol{x} - (\boldsymbol{x}_\lambda^\star - \frac{1}{L}\nabla f(\boldsymbol{x}_\lambda^\star))\|^2 \\ =&\Big(\frac{1}{L^2}\|\nabla f(\boldsymbol{x}_\lambda^\star)\|^2 - \frac{2}{L}\langle \nabla f(\boldsymbol{x}_\lambda^\star), \boldsymbol{x}_\lambda^\star \rangle \Big) \\ &+ \frac{2}{L}\Big(\langle \nabla f(\boldsymbol{x}_\lambda^\star), \boldsymbol{x} \rangle + \frac{L}{2}\|\boldsymbol{x} - \boldsymbol{x}_\lambda^\star\|^2 \Big), \end{aligned}$$

we obtain

$$\begin{aligned} &(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star) \\ &= \operatorname*{arg\,min}_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \Big\{ \frac{L}{2}\|\boldsymbol{x} - (\boldsymbol{x}_\lambda^\star - \frac{1}{L}\nabla f(\boldsymbol{x}_\lambda^\star))\|^2 + \lambda\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w}) \Big\}, \end{aligned}$$

which indicates (29) holds.

(ii) In the case of $(\boldsymbol{w}_\lambda^\star)_i = 0$, any $(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}$ should satisfy the optimality condition of problem (29) with respect to $\boldsymbol{x}$

$$L((\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i} - (\boldsymbol{z}_\lambda^\star)_{\mathbb{G}_i}) = 0.$$

Since $L > 0$, the solution of the above equation is $(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i} = (\boldsymbol{z}_\lambda^\star)_{\mathbb{G}_i}$.

In the case of $(\boldsymbol{w}_\lambda^\star)_i = 1$, according to the first-order optimality condition of problem (29) with respect to $\boldsymbol{x}$, any non-zero element $(\boldsymbol{x}_\lambda^\star)_j \in (\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}$ satisfies

$$\begin{aligned} &L((\boldsymbol{x}_\lambda^\star)_j - (\boldsymbol{z}_\lambda^\star)_j) + \\ &\lambda q(\|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\|)_p^{q-p}|(\boldsymbol{x}_\lambda^\star)_j|^{p-1} sign((\boldsymbol{x}_\lambda^\star)_j) = 0, \end{aligned} \tag{31}$$

Hence, in the case of $p \in \{1,2\}$ and $q = 1$, $(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star)$ satisfies (30).

■

Lemma 5: For any given λ ($\lambda \geq 0$), let $\mathbb{C}_\lambda^\star = \{\boldsymbol{x}_\lambda^\star\}$. In the case of $0 < p \leq 1, 0 < q \leq 1$, suppose

$\mathbb{C}_\lambda^\star$ is bounded. Let $\lambda_f = \max_{\boldsymbol{x}\in\mathbb{C}_\lambda^\star} \|\nabla f(\boldsymbol{x})\|_\infty$, and let $\alpha = \max_{i\in\{1,\ldots,N\},j\in\mathbb{G}_i} \frac{1}{q}\|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\|_p^{p-q}|(\boldsymbol{x}_\lambda^\star)_j|^{1-p}$, where $(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i} \neq 0$. Then in the case of $0 < p \leq 1, 0 < q \leq 1$, the following statements hold:

(i) $\alpha \neq \infty$.

(ii) For any $\lambda > \alpha\lambda_f$, $\psi_{p,q}(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star) = 0$.

Proof: (i) For a non-zero group $(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}$, let $i_{max} = \arg\max_{j\in\mathbb{G}_i} |(\boldsymbol{x}_\lambda^\star)_j|$, then we have

$$|(\boldsymbol{x}_\lambda^\star)_{i_{max}}| \leq \|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\|_p \leq (N_i)^{\frac{1}{p}}|(\boldsymbol{x}_\lambda^\star)_{i_{max}}|,$$

where $N_i = |\mathbb{G}_i| \geq 1$. Therefore, for any non-zero group $(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}, \forall j \in \mathbb{G}_i$, there holds

$$\begin{cases} \|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\|_p^{p-q}|(\boldsymbol{x}_\lambda^\star)_j|^{1-p} \leq & |N_i|^{\frac{p-q}{p}}|(\boldsymbol{x}_\lambda^\star)_{i_{max}}|^{1-q}, \\ & \text{if } 0 < q \leq p \leq 1; \\ \|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\|_p^{p-q}|(\boldsymbol{x}_\lambda^\star)_j|^{1-p} \leq & |(\boldsymbol{x}_\lambda^\star)_{i_{max}}|^{1-q}, \\ & \text{if } 0 < p < q \leq 1. \end{cases} \tag{32}$$

According to the assumption that $\mathbb{C}_\lambda^\star$ is bounded, we have $|(\boldsymbol{x}_\lambda^\star)_{i_{max}}| \neq \infty$. This together with (32) indicate that $\alpha \neq \infty$.

(ii) According to (31), we obtain

$$|(\nabla f(\boldsymbol{x}_\lambda^\star))_j| = \lambda q \|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\|_p^{q-p}|(\boldsymbol{x}_\lambda^\star)_j|^{p-1}. \tag{33}$$

Next we prove by contradiction. Suppose that there exists $\lambda > \alpha\lambda_f$ such that $\psi_{p,q}(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star) = \langle \boldsymbol{w}_\lambda^\star, [\boldsymbol{x}_\lambda^\star]_{p,q}\rangle \neq 0$. That means there exists $i \in \mathbb{S}(\boldsymbol{w}_\lambda^\star)$ and $L \geq L_f$ such that $(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i} \neq 0$. By (31), $\forall (\boldsymbol{x}_\lambda^\star)_j \neq 0 (j \in \mathbb{G}_i)$, there holds that

$$|(\nabla f(\boldsymbol{x}_\lambda^\star))_j| = \lambda q \|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\|_p^{q-p}|(\boldsymbol{x}_\lambda^\star)_j|^{p-1} > \frac{\lambda}{\alpha}, \quad j \in \mathbb{G}_i.$$

This implies that

$$|(\nabla f(\boldsymbol{x}_\lambda^\star))_j| > \frac{\lambda}{\alpha} > \lambda_f = \max_{\boldsymbol{x}\in\mathbb{C}_\lambda^\star} \|\nabla f(\boldsymbol{x})\|_\infty,$$

which contradicts the definition of $\|\nabla f(\boldsymbol{x})\|_\infty$. ■

Lemma 6: Let $N_{\max} = \max\{N_i, i = 1, \ldots, N\}$, $\lambda_f = \max_{\boldsymbol{x}\in\mathbb{C}_\lambda^\star} \|\nabla f(\boldsymbol{x})\|_\infty$. Then for any $\lambda > \sqrt{N_{\max}}\lambda_f$, $\psi_{2,1}(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star) = 0$.

Proof: We prove by contradiction. Suppose that there exists $\lambda > \sqrt{N_{max}}\lambda_f$ such that $\psi_{2,1}(\boldsymbol{x}_\lambda^\star, \boldsymbol{w}_\lambda^\star) =< \boldsymbol{w}_\lambda^\star, [\boldsymbol{x}_\lambda^\star]_{2,1} >\neq 0$. That is, by (30), there exists $i \in \mathbb{S}(\boldsymbol{w}_\lambda^\star)$ such that $(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i} = soft((\boldsymbol{z}_\lambda^\star)_{\mathbb{G}_i}) \neq 0$, i.e.,

$$(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i} = (\|(\boldsymbol{z}_\lambda^\star)_{\mathbb{G}_i}\| - \frac{\lambda}{L})\frac{(\boldsymbol{z}_\lambda^\star)_{\mathbb{G}_i}}{\|(\boldsymbol{z}_\lambda^\star)_{\mathbb{G}_i}\|} \neq 0.$$

In such a case, it holds that

$$\|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\| = \|(\boldsymbol{z}_\lambda^\star)_{\mathbb{G}_i}\| - \frac{\lambda}{L} = \|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i} - \frac{1}{L}(\nabla f(\boldsymbol{x}_\lambda^\star))_{\mathbb{G}_i}\| - \frac{\lambda}{L}.$$

Then we obtain

$$\|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\| \leq \|(\boldsymbol{x}_\lambda^\star)_{\mathbb{G}_i}\| + \frac{1}{L}\|(\nabla f(\boldsymbol{x}_\lambda^\star))_{\mathbb{G}_i}\| - \frac{\lambda}{L},$$

from which we can get that

$$\|(\nabla f(\boldsymbol{x}_\lambda^\star))_{\mathbb{G}_i}\| \geq \lambda > \sqrt{N_{max}}\lambda_f.$$

According to the definition of λ_f, there exists $j \in \mathbb{G}_i$ such that $|(\nabla f(\boldsymbol{x}_\lambda^\star))_j| > \max_{\boldsymbol{x}\in\mathbb{C}_\lambda^\star} \|\nabla f(\boldsymbol{x})\|_\infty$, which contradicts the definition of $\|\nabla f(\boldsymbol{x})\|_\infty$. ■

Next, we prove Theorem 1.

Proof: According to the assumption, $\{\nabla f(\boldsymbol{x}) : \boldsymbol{x} \in \mathbb{C}_\lambda^\star\}$ is bounded, we have $\lambda_f = \max_{\boldsymbol{x}\in\mathbb{C}_\lambda^\star} \|\nabla f(\boldsymbol{x})\|_\infty \neq \infty$. Then by Lemma 5 and Lemma 6, in the case of $0 < p \leq 1, 0 < q \leq 1$, or $p = 2, q = 1$, there exists $\bar{\lambda} \neq \infty$ such that $\psi_{p,q}(\boldsymbol{x}_{\bar{\lambda}}^\star, \boldsymbol{w}_{\bar{\lambda}}^\star) = 0$, where $(\boldsymbol{x}_{\bar{\lambda}}^\star, \boldsymbol{w}_{\bar{\lambda}}^\star)$ is an optimal solution of $g(\bar{\lambda})$. Combining the above conclusion with Lemma 1, we can get that $\boldsymbol{x}_{\bar{\lambda}}^\star$ is a feasible solution of problem (1). Then it holds that

$$\max_{\lambda\geq 0} \min_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \{f(\boldsymbol{x}) + \lambda\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w})\} \geq f(\boldsymbol{x}_{\bar{\lambda}}^\star) \geq \min_{\boldsymbol{x}\in\mathbb{C}_s} f(\boldsymbol{x}).$$

Further, according to the weak duality theorem, it holds

$$\min_{\boldsymbol{x}\in\mathbb{C}_s} f(\boldsymbol{x}) \geq \max_{\lambda\geq 0} \min_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \{f(\boldsymbol{x}) + \lambda\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w})\}.$$

So the strong duality property holds. ■

Appendix C
Proof of Lemma 2

Proof: According to (28), we set $\boldsymbol{w}_i = 0, \forall i \in \mathbb{S}_\mathbb{G}(\boldsymbol{x})$. In the case of $\boldsymbol{w}_i = 0$, any $\boldsymbol{x}_{\mathbb{G}_i}$ which is a minimizer of problem (15) should satisfy the optimality condition with respect to $\boldsymbol{x}$:

$$L_k(\boldsymbol{x}_{\mathbb{G}_i} - \boldsymbol{z}_{\mathbb{G}_i}^k) = 0. \tag{34}$$

Since $L_k > 0$, the solution of the above equation is $\boldsymbol{x}_{\mathbb{G}_i} = \boldsymbol{z}_{\mathbb{G}_i}^k$ and consequently the minimum value of problem (15) is $\mathcal{Y}_{L_k,\lambda,p,q,\boldsymbol{z}_{\mathbb{G}_i}^k}(\boldsymbol{z}_{\mathbb{G}_i}^k, 0) = 0$.

In the case of $\boldsymbol{w}_i = 1$, according to the first-order optimality condition of problem (15) with respect to $\boldsymbol{x}$, any non-zero element in solution should satisfy

$$L_k(\boldsymbol{x}_j - \boldsymbol{z}_j^k) + \lambda q(\|\boldsymbol{x}_{\mathbb{G}_i}\|)_p^{q-p}|\boldsymbol{x}_j|^{p-1}\operatorname{sign}(\boldsymbol{x}_j) = 0, \ \forall j \in \mathbb{G}_i.$$

Hence, in the case of $p \in \{1, 2\}$ and $q = 1$, we obtain the solution $\boldsymbol{x}_{\mathbb{G}_i} = \operatorname{soft}_{\frac{\lambda}{L_k},p}(\boldsymbol{z}_{\mathbb{G}_i}^k)$. ■

Appendix D
Proof of Theorem 2

First, we introduce a lemma.

Lemma 7: Given scalars y_1 and y_2. If $|y_1| \geq |y_2|$, then $|\operatorname{soft}_{\frac{\lambda}{L},1}(y_1) - y_1| \geq |\operatorname{soft}_{\frac{\lambda}{L},1}(y_2) - y_2|$, where $\operatorname{soft}_{\frac{\lambda}{L},1}(\cdot)$ is the soft-thresholding operator defined as (5).

Proof: For simplicity, we let $\operatorname{soft}(\cdot)$ be the abbreviation of $\operatorname{soft}_{\frac{\lambda}{L},1}(\cdot)$ in the following proof, i.e., for a given λ and L, the soft-thresholding operator for a scalar y can be written as:

$$\operatorname{soft}(y) = \operatorname{sign}(y)\max(|y| - \frac{\lambda}{L}, 0). \tag{35}$$

Given $|y_1| \geq |y_2|$, we prove Lemma 7 by considering the following cases:

(i) $\operatorname{soft}(y_1) = \operatorname{soft}(y_2) = 0$. Obviously, in this situation, $|\operatorname{soft}(y_1) - y_1| = |y_1| \geq |y_2| = |\operatorname{soft}(y_2) - y_2|$.

(ii) $\operatorname{soft}(y_1) = 0, \operatorname{soft}(y_2) \neq 0$. According to the definition of $\operatorname{soft}(\cdot)$ in (35), there does not exist such a case.

(iii) $\operatorname{soft}(y_1) \neq 0, \operatorname{soft}(y_2) \neq 0$. According to (35), we have $|\operatorname{soft}(y_1) - y_1| = |\operatorname{soft}(y_2) - y_2| = \frac{\lambda}{L}$.

(iv) $\operatorname{soft}(y_1) \neq 0, \operatorname{soft}(y_2) = 0$. According to (35), we have $|\operatorname{soft}(y_1) - y_1| = \frac{\lambda}{L}$ and $|\operatorname{soft}(y_2) - y_2| \leq \frac{\lambda}{L}$, which means $|\operatorname{soft}(y_1) - y_1| \geq |\operatorname{soft}(y_2) - y_2|$.

By the above analysis, we have Lemma 7 holds. ■

Next, we prove Theorem 2. For simplicity, we let $\mathcal{Y}_p(\boldsymbol{x}_{\mathbb{G}_i}, \boldsymbol{w}_i), \operatorname{soft}_p(\cdot)$ be the abbreviation of $\mathcal{Y}_{L_k,\lambda,p,1,\boldsymbol{z}^k_{\mathbb{G}_i}}(\boldsymbol{x}_{\mathbb{G}_i}, \boldsymbol{w}_i), \operatorname{soft}_{\frac{\lambda}{L_k},p}(\cdot)$ in the following proof, respectively.

Proof: According to (16), in this proof we define $(\boldsymbol{x}^\star, \boldsymbol{w}^\star)$ as

$$(\boldsymbol{x}^\star_{\mathbb{G}_i}, \boldsymbol{w}^\star_i) = \begin{cases} (\boldsymbol{z}^k_{\mathbb{G}_i}, 0) & \text{if } i \in \mathbb{A}_s(\boldsymbol{z}^k); \\ (\operatorname{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) & \text{otherwise.} \end{cases}$$

Then we can obtain that

$$\|\boldsymbol{z}^k_{\mathbb{G}_i}\| \geq \|\boldsymbol{z}^k_{\mathbb{G}_j}\| \quad \forall i \in \mathbb{S}^c(\boldsymbol{w}^\star), \forall j \in \mathbb{S}(\boldsymbol{w}^\star). \tag{36}$$

Also, we denote a feasible solution of problem (14) by $(\boldsymbol{x}, \boldsymbol{w})$.

(i) By (6) we have

$$\mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) = \begin{cases} \lambda(\|\boldsymbol{z}^k_{\mathbb{G}_i}\| - \frac{\lambda}{2L_k}) & \text{if } \|\boldsymbol{z}^k_{\mathbb{G}_i}\| > \frac{\lambda}{L_k}; \\ \frac{L_k}{2}\|\boldsymbol{z}^k_{\mathbb{G}_i}\|^2 & \text{if } \|\boldsymbol{z}^k_{\mathbb{G}_i}\| \leq \frac{\lambda}{L_k}. \end{cases}$$

Hence we can get the following analyses:

1) If $\|\boldsymbol{z}^k_{\mathbb{G}_i}\| = 0$, then $\mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) = 0$.
2) If $\|\boldsymbol{z}^k_{\mathbb{G}_i}\| \geq \|\boldsymbol{z}^k_{\mathbb{G}_j}\|$ $(\forall i, j \in 1, \ldots, N)$, then $\mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) \geq \mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_j}), 1)$.
3) If $|\mathbb{S}^c(\boldsymbol{w}^\star)| < s$, then $\forall j \in \mathbb{S}(\boldsymbol{w}^\star)$, $\|\boldsymbol{z}^k_{\mathbb{G}_j}\| = 0$ and $\mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_j}), 1) = 0$.
4) If $|\mathbb{S}^c(\boldsymbol{w}^\star)| = s$, then $|\mathbb{S}^c(\boldsymbol{w})| \leq s = |\mathbb{S}^c(\boldsymbol{w}^\star)|$ implies that

$$\begin{aligned} |\mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}(\boldsymbol{w})| &= |\mathbb{S}^c(\boldsymbol{w}^\star)| - |\mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}^c(\boldsymbol{w})| \\ &\geq |\mathbb{S}^c(\boldsymbol{w})| - |\mathbb{S}^c(\boldsymbol{w}) \cap \mathbb{S}^c(\boldsymbol{w}^\star)| \\ &= |\mathbb{S}^c(\boldsymbol{w}) \cap \mathbb{S}(\boldsymbol{w}^\star)|. \end{aligned} \tag{37}$$

Combining the above analyses and inequalities (36) and (37), the following inequality holds:

$$\sum_{i \in \mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}(\boldsymbol{w})} \mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) \geq \sum_{i \in \mathbb{S}(\boldsymbol{w}^\star) \cap \mathbb{S}^c(\boldsymbol{w})} \mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1). \tag{38}$$

Then,

$$\begin{aligned} &\mathcal{Y}_2(\boldsymbol{x}, \boldsymbol{w}) - \mathcal{Y}_2(\boldsymbol{x}^\star, \boldsymbol{w}^\star) \\ =&\Big(\sum_{i \in \mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}^c(\boldsymbol{w})} + \sum_{i \in \mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}(\boldsymbol{w})} + \sum_{i \in \mathbb{S}(\boldsymbol{w}^\star) \cap \mathbb{S}^c(\boldsymbol{w})} + \\ &\sum_{i \in \mathbb{S}(\boldsymbol{w}^\star) \cap \mathbb{S}(\boldsymbol{w})}\Big)(\mathcal{Y}_2(\boldsymbol{x}_{\mathbb{G}_i}, \boldsymbol{w}_i) - \mathcal{Y}_2(\boldsymbol{x}^\star_{\mathbb{G}_i}, \boldsymbol{w}^\star_i)) \\ =&\sum_{i \in \mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}^c(\boldsymbol{w})} (\mathcal{Y}_2(\boldsymbol{x}_{\mathbb{G}_i}, 0) - \mathcal{Y}_2(\boldsymbol{z}^k_{\mathbb{G}_i}, 0)) + \\ &\sum_{i \in \mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}(\boldsymbol{w})} (\mathcal{Y}_2(\boldsymbol{x}_{\mathbb{G}_i}, 1) - \mathcal{Y}_2(\boldsymbol{z}^k_{\mathbb{G}_i}, 0)) + \\ &\sum_{i \in \mathbb{S}(\boldsymbol{w}^\star) \cap \mathbb{S}^c(\boldsymbol{w})} (\mathcal{Y}_2(\boldsymbol{x}_{\mathbb{G}_i}, 0) - \mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1)) + \\ &\sum_{i \in \mathbb{S}(\boldsymbol{w}^\star) \cap \mathbb{S}(\boldsymbol{w})} (\mathcal{Y}_2(\boldsymbol{x}_{\mathbb{G}_i}, 1) - \mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1)) \\ \geq&\sum_{i \in \mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}(\boldsymbol{w})} (\mathcal{Y}_2(\boldsymbol{x}_{\mathbb{G}_i}, 1) - \mathcal{Y}_2(\boldsymbol{z}^k_{\mathbb{G}_i}, 0)) + \\ &\sum_{i \in \mathbb{S}(\boldsymbol{w}^\star) \cap \mathbb{S}^c(\boldsymbol{w})} (\mathcal{Y}_2(\boldsymbol{x}_{\mathbb{G}_i}, 0) - \mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1)) \\ \geq&\sum_{i \in \mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}(\boldsymbol{w})} (\mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) - \mathcal{Y}_2(\boldsymbol{z}^k_{\mathbb{G}_i}, 0)) + \\ &\sum_{i \in \mathbb{S}(\boldsymbol{w}^\star) \cap \mathbb{S}^c(\boldsymbol{w})} (\mathcal{Y}_2(\boldsymbol{z}^k_{\mathbb{G}_i}, 0) - \mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1)) \\ =&\sum_{i \in \mathbb{S}^c(\boldsymbol{w}^\star) \cap \mathbb{S}(\boldsymbol{w})} \mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) - \\ &\sum_{i \in \mathbb{S}(\boldsymbol{w}^\star) \cap \mathbb{S}^c(\boldsymbol{w})} \mathcal{Y}_2(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) \\ \geq&0 \end{aligned} \tag{39}$$

where the last inequality comes from the definition of $\mathbb{S}^c(\boldsymbol{w}^\star)$ and inequality (38). Hence, $(\boldsymbol{x}^\star, \boldsymbol{w}^\star)$ is an optimal solution of problem (14).

(ii) If there is only one element in each group, then $\boldsymbol{x}_{\mathbb{G}_i} = \boldsymbol{x}_i, \boldsymbol{z}_{\mathbb{G}_i} = \boldsymbol{z}_i$, $N = n$, and we have

$$\begin{aligned} \mathcal{Y}_1(\boldsymbol{x}_{\mathbb{G}_i}, \boldsymbol{w}_i) &= \frac{L_k}{2}\|\boldsymbol{x}_{\mathbb{G}_i} - \boldsymbol{z}^k_{\mathbb{G}_i}\|^2 + \lambda(\boldsymbol{w}_i \times \|\boldsymbol{x}_{\mathbb{G}_i}\|_1) \\ &= \frac{L_k}{2}\|\boldsymbol{x}_i - \boldsymbol{z}^k_i\|^2 + \lambda(\boldsymbol{w}_i \times |\boldsymbol{x}_i|). \end{aligned}$$

For any $i, j \in \{1, 2, \ldots, N\})$, if $|\boldsymbol{z}_i| \geq |\boldsymbol{z}_j|$, by Lemma 7 we have $|\operatorname{soft}_1(\boldsymbol{z}_i) - \boldsymbol{z}_i| \geq |\operatorname{soft}_1(\boldsymbol{z}_j) - \boldsymbol{z}_j|$. Hence there holds

$$\begin{aligned} &\frac{L_k}{2}\|\operatorname{soft}_1(\boldsymbol{z}^k_i) - \boldsymbol{z}^k_i\|^2 + \lambda|\operatorname{soft}_1(\boldsymbol{z}^k_i)| \\ \geq&\frac{L_k}{2}\|\operatorname{soft}_1(\boldsymbol{z}^k_j) - \boldsymbol{z}^k_j\|^2 + \lambda|\operatorname{soft}_1(\boldsymbol{z}^k_j)|, \end{aligned}$$

i.e., $\mathcal{Y}_1(soft(\boldsymbol{z}^k_i), 1) \geq \mathcal{Y}_1(soft(\boldsymbol{z}^k_j), 1)$.

Similar to the proof in inequality (39), we can get $\mathcal{Y}_1(\boldsymbol{x}, \boldsymbol{w}) - \mathcal{Y}_1(\boldsymbol{x}^\star, \boldsymbol{w}^\star) \geq 0$. Hence, $(\boldsymbol{x}^\star, \boldsymbol{w}^\star)$ is an optimal solution of problem (14). ■

APPENDIX E
PROOF OF THEOREM 3

Proof: Problem (14) can be decomposed into problem (15), which has a closed-form solution under specific (p, q) values. For simplicity, here we let $\mathcal{Y}_p(\boldsymbol{x}_{\mathbb{G}_i}, \boldsymbol{w}_i), \operatorname{soft}_p(\cdot)$ be the abbreviation of $\mathcal{Y}_{L_k,\lambda,p,1,\boldsymbol{z}^k_{\mathbb{G}_i}}(\boldsymbol{x}_{\mathbb{G}_i}, \boldsymbol{w}_i), \operatorname{soft}_{\frac{\lambda}{L_k},p}(\cdot)(p \in \{1, 2\})$ respectively.

For $(p, q) = (1, 1)$ and $(p, q) = (2, 1)$, we denote the solution by $(\boldsymbol{x}^\star_{\mathbb{G}_i}, \boldsymbol{w}^\star_i)$, i.e.,

$$(\boldsymbol{x}^\star_{\mathbb{G}_i}, \boldsymbol{w}^\star_i) = \begin{cases} (\boldsymbol{z}^k_{\mathbb{G}_i}, 0) & \text{if } i \in \mathbb{B}_s(\boldsymbol{z}^k); \\ (\operatorname{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) & \text{otherwise.} \end{cases}$$

According to the definition of $\mathbb{B}_s(\boldsymbol{z}^k)$, we have

$$\mathcal{Y}_p(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) \geq \mathcal{Y}_p(\operatorname{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_j}), 1), \quad \forall i \in \mathbb{S}^c(\boldsymbol{w}^\star), \forall j \in \mathbb{S}(\boldsymbol{w}^\star). \tag{40}$$

Then we can get the following results:

1) If $|\mathbb{S}^c(\boldsymbol{w}^\star)| < s$, then $\forall j \in \mathbb{S}(\boldsymbol{w}^\star)$, $\mathcal{Y}_2(\text{soft}_2(\boldsymbol{z}^k_{\mathbb{G}_j}), 1) = 0$.
2) If $|\mathbb{S}^c(\boldsymbol{w}^\star)| = s$, then (37) holds.

Suppose that $(\boldsymbol{x}, \boldsymbol{w})$ is a feasible solution of problem (14). Then we have

$$\begin{aligned}
&\mathcal{Y}_p(\boldsymbol{x}, \boldsymbol{w}) - \mathcal{Y}_p(\boldsymbol{x}^\star, \boldsymbol{w}^\star)\\
=&(\sum_{i\in\mathbb{S}^c(\boldsymbol{w}^\star)\cap\mathbb{S}^c(\boldsymbol{w})} + \sum_{i\in\mathbb{S}^c(\boldsymbol{w}^\star)\cap\mathbb{S}(\boldsymbol{w})} + \sum_{i\in\mathbb{S}(\boldsymbol{w}^\star)\cap\mathbb{S}^c(\boldsymbol{w})} +\\
&\sum_{i\in\mathbb{S}(\boldsymbol{w}^\star)\cap\mathbb{S}(\boldsymbol{w})})(\mathcal{Y}_p(\boldsymbol{x}_{\mathbb{G}_i}, \boldsymbol{w}_i) - \mathcal{Y}_p(\boldsymbol{x}^\star_{\mathbb{G}_i}, \boldsymbol{w}^\star_i))\\
=& \sum_{i\in\mathbb{S}^c(\boldsymbol{w}^\star)\cap\mathbb{S}^c(\boldsymbol{w})} (\mathcal{Y}_p(\boldsymbol{x}_{\mathbb{G}_i}, 0) - \mathcal{Y}_p(\boldsymbol{z}^k_{\mathbb{G}_i}, 0)) +\\
&\sum_{i\in\mathbb{S}^c(\boldsymbol{w}^\star)\cap\mathbb{S}(\boldsymbol{w})} (\mathcal{Y}_p(\boldsymbol{x}_{\mathbb{G}_i}, 1) - \mathcal{Y}_p(\boldsymbol{z}^k_{\mathbb{G}_i}, 0)) +\\
&\sum_{i\in\mathbb{S}(\boldsymbol{w}^\star)\cap\mathbb{S}^c(\boldsymbol{w})} (\mathcal{Y}_p(\boldsymbol{x}_{\mathbb{G}_i}, 0) - \mathcal{Y}_p(\text{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1)) +\\
&\sum_{i\in\mathbb{S}(\boldsymbol{w}^\star)\cap\mathbb{S}(\boldsymbol{w})} (\mathcal{Y}_p(\boldsymbol{x}_{\mathbb{G}_i}, 1) - \mathcal{Y}_p(\text{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1))\\
\geq& \sum_{i\in\mathbb{S}^c(\boldsymbol{w}^\star)\cap\mathbb{S}(\boldsymbol{w})} (\mathcal{Y}_p(\boldsymbol{x}_{\mathbb{G}_i}, 1) - \mathcal{Y}_p(\boldsymbol{z}^k_{\mathbb{G}_i}, 0)) +\\
&\sum_{i\in\mathbb{S}(\boldsymbol{w}^\star)\cap\mathbb{S}^c(\boldsymbol{w})} (\mathcal{Y}_p(\boldsymbol{x}_{\mathbb{G}_i}, 0) - \mathcal{Y}_p(\text{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1))\\
\geq& \sum_{i\in\mathbb{S}^c(\boldsymbol{w}^\star)\cap\mathbb{S}(\boldsymbol{w})} (\mathcal{Y}_p(\text{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) - \mathcal{Y}_p(\boldsymbol{z}^k_{\mathbb{G}_i}, 0)) +\\
&\sum_{i\in\mathbb{S}(\boldsymbol{w}^\star)\cap\mathbb{S}^c(\boldsymbol{w})} (\mathcal{Y}_p(\boldsymbol{z}^k_{\mathbb{G}_i}, 0) - \mathcal{Y}_p(\text{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1))\\
=& \sum_{i\in\mathbb{S}^c(\boldsymbol{w}^\star)\cap\mathbb{S}(\boldsymbol{w})} (\mathcal{Y}(\text{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) - \mathcal{Y}_p(\boldsymbol{z}^k_{\mathbb{G}_i}, 0)) -\\
&\sum_{i\in\mathbb{S}(\boldsymbol{w}^\star)\cap\mathbb{S}^c(\boldsymbol{w})} (\mathcal{Y}_p(\text{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) - \mathcal{Y}_p(\boldsymbol{z}^k_{\mathbb{G}_i}, 0))\\
=& \sum_{i\in\mathbb{S}^c(\boldsymbol{w}^\star)\cap\mathbb{S}(\boldsymbol{w})} \mathcal{Y}_p(\text{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1) -\\
&\sum_{i\in\mathbb{S}(\boldsymbol{w}^\star)\cap\mathbb{S}^c(\boldsymbol{w})} \mathcal{Y}_p(\text{soft}_p(\boldsymbol{z}^k_{\mathbb{G}_i}), 1)\\
\geq& 0,
\end{aligned}$$

where the last inequality comes from inequalities (37) and (40). Hence, $(\boldsymbol{x}^\star, \boldsymbol{w}^\star)$ is an optimal solution of problem (14). ■

Appendix F
Proof of Lemma 3

Inspired by [1], we have the following proof.

Proof: (i) In the following proof, $\varsigma \in (0,1)$ is a small positive constant. And for simplicity, we let $\mathcal{L}, \psi$ be the abbreviation of $\mathcal{L}_{p,q}, \psi_{p,q}$ $(p \in \{1,2\}, q = 1)$ respectively.

If $(\mathbb{S}^{k+1}_{\mathbb{G}}$, p, $q)$ matches any of the optimal settings in Definition 4, according to line 6 of Algorithm 1, there holds

$$(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) = T_{L_k,\lambda,p}(\boldsymbol{x}^k) = \operatorname*{arg\,min}_{(\boldsymbol{x},\boldsymbol{w})\in\Omega} \mathcal{P}_{L_k,\lambda,p,q,\boldsymbol{x}^k}(\boldsymbol{x}, \boldsymbol{w}),$$

which means $(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1})$ achieves a better value of $\mathcal{P}_{L_k,\lambda,p,q,\boldsymbol{x}^k}(\boldsymbol{x}, \boldsymbol{w})$ than $(\boldsymbol{x}^k, \boldsymbol{w}^k)$. So we get the inequality

$$\mathcal{P}_{L_k,\lambda,p,q,\boldsymbol{x}^k}(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) \leq \mathcal{P}_{L_k,\lambda,p,q,\boldsymbol{x}^k}(\boldsymbol{x}^k, \boldsymbol{w}^k),$$

i.e.,

$$\begin{aligned}
&\left(\langle \nabla f(\boldsymbol{x}^k), \boldsymbol{x}^{k+1} - \boldsymbol{x}^k\rangle + \frac{L_k}{2}\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2\right)\\
&\qquad + \lambda\psi(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) \leq \lambda\psi(\boldsymbol{x}^k, \boldsymbol{w}^k).
\end{aligned}\tag{41}$$

Further, since $\nabla f(\boldsymbol{x})$ is Lipschitz continuous with constant L_f, it holds that

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}, \lambda) = f(\boldsymbol{x}^{k+1}) + \lambda\psi(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1})\\
&\leq (f(\boldsymbol{x}^k) + \langle \nabla f(\boldsymbol{x}^k), \boldsymbol{x}^{k+1} - \boldsymbol{x}^k\rangle + \frac{L_f}{2}\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2)\\
&+ \lambda\psi(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}).
\end{aligned}\tag{42}$$

Combining (41) with (42), we obtain

$$\mathcal{L}(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}, \lambda) \leq \mathcal{L}(\boldsymbol{x}^k, \boldsymbol{w}^k, \lambda) - \frac{L_k - L_f}{2}\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2.$$

Therefore, the monotone line search stopping criterion, i.e.,

$$\mathcal{L}(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}, \lambda) \leq \mathcal{L}(\boldsymbol{x}^k, \boldsymbol{w}^k, \lambda) - \frac{\varsigma}{2}L_k\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2, \tag{43}$$

is satisfied whenever $\frac{L_k - L_f}{2} \geq \frac{\varsigma L_k}{2}$, i.e., $L_k \geq \frac{L_f}{1-\varsigma}$.

Next, using the induction method we prove that the non-monotone line search stopping criterion (20) is also satisfied for all $k \geq 0$ whenever $L_k \geq \frac{L_f}{1-\varsigma}$.

According to the definition of c_k in (19), setting $\theta_0 = 1$ and $c_0 = \mathcal{L}(\boldsymbol{x}^0, \boldsymbol{w}^0, \lambda)$, c_k can be efficiently computed by the following recursion:

$$\theta_{k+1} = \gamma\theta_k + 1, \tag{44}$$

$$c_{k+1} = \frac{\gamma\theta_k c_k + \mathcal{L}(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}, \lambda)}{\theta_{k+1}}, \tag{45}$$

where $\gamma \in (0,1)$ is a given constant. Then for $k = 0$, by (43) we have that whenever $L_0 \geq \frac{L_f}{1-\varsigma}$ there holds

$$\mathcal{L}(\boldsymbol{x}^1, \boldsymbol{w}^1, \lambda) \leq c_0 - \frac{\varsigma}{2}L_0\|\boldsymbol{x}^1 - \boldsymbol{x}^0\|^2.$$

Assume that (20) holds for all $k = 0, 1, \ldots, j$, then we consider $k = j + 1$. Define

$$\mathcal{D}_{j+1}(t) = \frac{tc_j + \mathcal{L}(\boldsymbol{x}^{j+1}, \boldsymbol{w}^{j+1}, \lambda)}{t + 1},$$

then

$$\frac{d}{dt}\mathcal{D}_{j+1}(t) = \frac{c_j - \mathcal{L}(\boldsymbol{x}^{j+1}, \boldsymbol{w}^{j+1}, \lambda)}{(t+1)^2}.$$

According to the inductive hypothesis, it holds that

$$\mathcal{L}(\boldsymbol{x}^{j+1}, \boldsymbol{w}^{j+1}, \lambda) \leq c_j.$$

Then we have

$$\frac{d}{dt}\mathcal{D}_{j+1}(t) \geq 0.$$

which means that $\mathcal{D}_{j+1}(t)$ is non-decreasing. Hence

$$\mathcal{L}(\boldsymbol{x}^{j+1}, \boldsymbol{w}^{j+1}, \lambda) = \mathcal{D}_{j+1}(0) \leq \mathcal{D}_{j+1}(\gamma\theta_j) = c_{j+1}, \tag{46}$$

where $\gamma \in (0,1)$ and θ_j is defined as (44). Combining (46) with (43), whenever $L_{j+1} \geq \frac{L_f}{1-\varsigma}$, there holds

$$\begin{aligned} &\mathcal{L}(\boldsymbol{x}^{j+2}, \boldsymbol{w}^{j+2}, \lambda) \\ \leq &\mathcal{L}(\boldsymbol{x}^{j+1}, \boldsymbol{w}^{j+1}, \lambda) - \frac{\varsigma}{2} L_{j+1} \|\boldsymbol{x}^{j+2} - \boldsymbol{x}^{j+1}\|^2 \\ \leq & c_{j+1} - \frac{\varsigma}{2} L_{j+1} \|\boldsymbol{x}^{j+2} - \boldsymbol{x}^{j+1}\|^2, \end{aligned}$$

which implies that (20) holds for $k = j+1$.

(ii) By (20), (44) and (45), for $k \geq 0$ we have

$$\begin{aligned} c_{k+1} &= \frac{\gamma \theta_k c_k + \mathcal{L}_{p,q}(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}, \lambda)}{\theta_{k+1}} \\ &\leq \frac{\gamma \theta_k c_k + c_k - \frac{\varsigma}{2} L_k \|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2}{\theta_{k+1}} \\ &= c_k - \frac{\varsigma L_k}{2\theta_{k+1}} \|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2. \end{aligned} \tag{47}$$

Hence c_k is monotonically decreasing. By Assumption A_2 and the definition of $\mathcal{L}, \psi$ in (9), (10), we obtain that $\mathcal{L}$ is bounded below. This together with line search stopping criterion (20) imply that c_k is bounded below. So there exists a number $c^\star$ such that

$$\lim_{k \to \infty} c_k = c^\star. \tag{48}$$

Furthermore, from the definition of θ_k in (44) we get

$$\theta_{k+1} = 1 + \sum_{j=1}^{k+1} \gamma^j = \sum_{j=0}^{k+1} \gamma^j \leq \sum_{j=0}^{\infty} \gamma^j = \frac{1}{1-\gamma},$$

which together with (47) indicate that

$$\begin{aligned} \frac{\varsigma L_k}{2}(1-\gamma)\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2 &\leq \frac{\varsigma L_k}{2\theta_{k+1}} \|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2 \\ &\leq c_k - c_{k+1}. \end{aligned}$$

Combining the above inequality with (48) leads to

$$\lim_{k \to \infty} \|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\| = 0. \tag{49}$$

■

Appendix G
Proof of Theorem 4

Proof: (i) By Lemma 3, we have $\lim_{k \to \infty} \|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\| = 0$, which together with the boundness assumption of $\{\boldsymbol{x}^k\}$ indicate that the sequence $\{\boldsymbol{x}^k\}$ converges.

(ii) For the sequence $\{\boldsymbol{w}^k\}$, since $\boldsymbol{w} \in \{0,1\}^n$ is bounded, $\{\boldsymbol{w}^k\}$ has a subsequence $\{\boldsymbol{w}^{i_k}\}$ which is convergent, say converges to $\boldsymbol{w}^\star$.

(iii) By Lemma 2 and the first-order optimality condition with respect to $\boldsymbol{x}$, in line 6 of Algorithm 1, each estimation $(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1})$ should satisfy:

$$\begin{aligned} & L_k(\boldsymbol{x}^{k+1} - \boldsymbol{z}^k) + \lambda \partial_{\boldsymbol{x}} \psi_{p,q}(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) \\ = & L_k(\boldsymbol{x}^{k+1} - \boldsymbol{x}^k + \frac{1}{L_k} \nabla f(\boldsymbol{x}^k)) + \lambda \partial_{\boldsymbol{x}} \psi_{p,q}(\boldsymbol{x}^{k+1}, \boldsymbol{w}^{k+1}) \\ = & 0. \end{aligned}$$

Since $\{\boldsymbol{x}^k\}$ is convergent, $\{\boldsymbol{x}^{i_k}\}$ is convergent. For any accumulation point $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}})$ of $\{(\boldsymbol{x}^{i_k}, \boldsymbol{w}^{i_k})\}$, the above equation indicates that

$$\nabla f(\hat{\boldsymbol{x}}) + \lambda \partial_{\boldsymbol{x}} \psi_{p,q}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}}) = \partial_{\boldsymbol{x}} \mathcal{L}_{p,q}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}}, \lambda) = 0. \tag{50}$$

In addition, Algorithm 1 identifies the group support set $\mathbb{S}_{\mathbb{G}}$ under the maximum group-sparsity constraint s, which together with operator $T_{L_k,\lambda,p}(\cdot)$ guarantee $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}}) \in \Omega$. Then according to Definition 5, $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}})$ is a partial first-order stationary point of (11) with respect to $\boldsymbol{x}$. ■

Appendix H
Proof of Theorem 5

Proof: We prove by contradiction. Suppose there exists a $\lambda > \lambda^\star$ such that $g(\lambda) < g(\lambda^\star)$, i.e.,

$$f(\boldsymbol{x}^\star_\lambda) + \lambda \psi_{p,q}(\boldsymbol{x}^\star_\lambda, \boldsymbol{w}^\star_\lambda) < f(\boldsymbol{x}^\star_{\lambda^\star}) + \lambda^\star \psi_{p,q}(\boldsymbol{x}^\star_{\lambda^\star}, \boldsymbol{w}^\star_{\lambda^\star}).$$

According to the definition of $(\boldsymbol{x}^\star_{\lambda^\star}, \boldsymbol{w}^\star_{\lambda^\star})$, we have

$$f(\boldsymbol{x}^\star_{\lambda^\star}) + \lambda^\star \psi_{p,q}(\boldsymbol{x}^\star_{\lambda^\star}, \boldsymbol{w}^\star_{\lambda^\star}) \leq f(\boldsymbol{x}^\star_\lambda) + \lambda^\star \psi_{p,q}(\boldsymbol{x}^\star_\lambda, \boldsymbol{w}^\star_\lambda).$$

The above two inequalities imply that

$$(\lambda - \lambda^\star) \psi_{p,q}(\boldsymbol{x}^\star_\lambda, \boldsymbol{w}^\star_\lambda) < 0.$$

Since $\lambda > \lambda^\star$, we have $\psi_{p,q}(\boldsymbol{x}^\star_\lambda, \boldsymbol{w}^\star_\lambda) < 0$, which contradicts the definition of $\psi_{p,q}$, i.e., $\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w}) =< \boldsymbol{w}, [\boldsymbol{x}_{\mathbb{G}}]_{p,q} >= \sum_{i=1}^N (w_i \times \|\boldsymbol{x}_{\mathbb{G}_i}\|_p^q)$. Hence $\forall \lambda > \lambda^\star$, $g(\lambda) \geq g(\lambda^\star)$, which together with the definition of $\lambda^\star$ indicate that $\forall \lambda > \lambda^\star$, $g(\lambda) = g(\lambda^\star)$.

In addition, it is well established that the Lagrange dual function $g(\lambda)$ is concave. This together with the above conclusion tells that $g(\lambda)$ is a non-decreasing function with respect to λ.

■

Appendix I
Proof of Theorem 6

First, we introduce some lemmas.

Lemma 8: If $\lambda_2 > \lambda_1 \geq 0$, then $\psi_{p,q}(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) \geq \psi_{p,q}(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2})$ and $f(\boldsymbol{x}^\star_{\lambda_1}) \leq f(\boldsymbol{x}^\star_{\lambda_2})$.

Proof: Since the following proof is independent of the value of (p,q), we let $\psi(\cdot)$ be the abbreviation of $\psi_{p,q}(\cdot)$. We prove the first inequality by contradiction. Suppose that there exist $\lambda_2 > \lambda_1 \geq 0$ such that $\psi(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) > \psi(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1})$, which means that there exists $\delta > 0$ such that $\psi(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) = \psi(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) + \delta$. Combining this assumption with (13), we get

$$\begin{aligned} f(\boldsymbol{x}^\star_{\lambda_2}) + \lambda_2 \psi(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) &= f(\boldsymbol{x}^\star_{\lambda_2}) + \lambda_2(\psi(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) + \delta) \\ &\leq f(\boldsymbol{x}^\star_{\lambda_1}) + \lambda_2 \psi(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}), \end{aligned}$$

and

$$\begin{aligned} f(\boldsymbol{x}^\star_{\lambda_1}) + \lambda_1 \psi(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) &\leq f(\boldsymbol{x}^\star_{\lambda_2}) + \lambda_1 \psi(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) \\ &= f(\boldsymbol{x}^\star_{\lambda_2}) + \lambda_1 \psi(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1} + \delta). \end{aligned}$$

From the above two inequalities we can obtain that

$$f(\boldsymbol{x}^\star_{\lambda_2}) + \lambda_2 \delta \leq f(\boldsymbol{x}^\star_{\lambda_1}) \leq f(\boldsymbol{x}^\star_{\lambda_2}) + \lambda_1 \delta,$$

which implies $\lambda_2 \leq \lambda_1$. This contradicts the assumption that $\lambda_2 > \lambda_1 \geq 0$. Hence, when $\lambda_2 > \lambda_1 \geq 0$, we have $\psi(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) \geq \psi(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2})$.

Furthermore, by the definition of $(\boldsymbol{x}^\star_{\lambda}, \boldsymbol{w}^\star_{\lambda})$ given in (13) we have

$$f(\boldsymbol{x}^\star_{\lambda_1}) - f(\boldsymbol{x}^\star_{\lambda_2}) \leq \lambda_1(\psi(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) - \psi(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1})).$$

Combining $\lambda_1 \geq 0$ with $\psi(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) \geq \psi(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2})$, we get $f(\boldsymbol{x}^\star_{\lambda_1}) \leq f(\boldsymbol{x}^\star_{\lambda_2})$. ■

Lemma 9: Suppose for some $\lambda_1 \geq 0$, $\psi_{p,q}(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) = 0$. Then for any $\lambda_2 > \lambda_1$, $\psi_{p,q}(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) = 0$ and $f(\boldsymbol{x}^\star_{\lambda_1}) = f(\boldsymbol{x}^\star_{\lambda_2})$.

Proof: By Lemma 8, we obtain that for any $\lambda_2 > \lambda_1$, $\psi_{p,q}(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) \leq \psi_{p,q}(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1})$. Together with the definition of $\psi_{p,q}(\boldsymbol{x}, \boldsymbol{w})$ in (10), it holds that $\psi_{p,q}(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) = 0$.

By (13), we have

$$\begin{cases} f(\boldsymbol{x}^\star_{\lambda_2}) + \lambda_2 \psi_{p,q}(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) \leq f(\boldsymbol{x}^\star_{\lambda_1}) + \lambda_2 \psi_{p,q}(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}); \\ f(\boldsymbol{x}^\star_{\lambda_1}) + \lambda_1 \psi_{p,q}(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) \leq f(\boldsymbol{x}^\star_{\lambda_2}) + \lambda_1 \psi_{p,q}(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}). \end{cases}$$

The above two inequalities together with $\psi_{p,q}(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) = \psi_{p,q}(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) = 0$ imply that $f(\boldsymbol{x}^\star_{\lambda_1}) = f(\boldsymbol{x}^\star_{\lambda_2})$. ■

Next, we prove Theorem 6.

Proof: We prove this by contradiction. Suppose there exists a $\lambda_1 \notin \mathbb{J}$ such that $\psi_{p,q}(\boldsymbol{x}^\star_{\lambda_1}, \boldsymbol{w}^\star_{\lambda_1}) = 0$. Then by Lemma 9, for any $\lambda_2 > \lambda_1$, $\psi_{p,q}(\boldsymbol{x}^\star_{\lambda_2}, \boldsymbol{w}^\star_{\lambda_2}) = 0$ and $f(\boldsymbol{x}^\star_{\lambda_1}) = f(\boldsymbol{x}^\star_{\lambda_2})$, which imply that $g(\lambda_1) = g(\lambda_2)$. Combing the above conclusion with Theorem 5, we have $g(\lambda_1) = \max\{g(\lambda)\}$, i.e., $\lambda_1 \in \mathbb{J}$, which contradicts the assumption. ■

Appendix J
Proof of Theorem 7

Proof: (i) Let $\boldsymbol{z}^\star = \boldsymbol{x}^\star - \frac{1}{L}\nabla f(\boldsymbol{x}^\star)$ where $1/L$ $(L > 0)$ is the selected step-size. If $\psi_{p,1}(\boldsymbol{x}^\star, \boldsymbol{w}^\star) =< \boldsymbol{w}^\star, [\boldsymbol{x}^\star]_{p,1} >= 0$, which combined with (16) imply that

$$\begin{cases} \boldsymbol{x}^\star_{\mathbb{G}_i} = \boldsymbol{z}^\star_{\mathbb{G}_i} = \boldsymbol{x}^\star_{\mathbb{G}_i} - \frac{1}{L}(\nabla f(\boldsymbol{x}^\star))_{\mathbb{G}_i} & \text{if } i \in \mathbb{S}^c(\boldsymbol{w}^\star); \\ \boldsymbol{x}^\star_{\mathbb{G}_i} = \mathrm{soft}_p(\boldsymbol{z}^\star_{\mathbb{G}_i}) = 0 & \text{if } i \in \mathbb{S}(\boldsymbol{w}^\star). \end{cases} \tag{51}$$

Hence, if $i \in \mathbb{S}^c(\boldsymbol{w}^\star)$, then $(\nabla f(\boldsymbol{x}^\star))_{\mathbb{G}_i} = 0$. And by the definition of $\mathrm{soft}_p(\cdot)$ in (5), (6), if $\boldsymbol{x}^\star_{\mathbb{G}_i} = 0$, then $\forall j \in \mathbb{G}_i, i \in \mathbb{S}(\boldsymbol{w}^\star)$,

$$\begin{cases} |z^\star_j| = |\frac{1}{L}(\nabla f(\boldsymbol{x}^\star))_j| \leq \frac{\lambda}{L} & \text{if } p = 1; \\ \|\boldsymbol{z}^\star_{\mathbb{G}_i}\| = \|\frac{1}{L}(\nabla f(\boldsymbol{x}^\star))_{\mathbb{G}_i}\| \leq \frac{\lambda}{L} & \text{if } p = 2. \end{cases}$$

This combined with (51) imply that the conclusion holds.

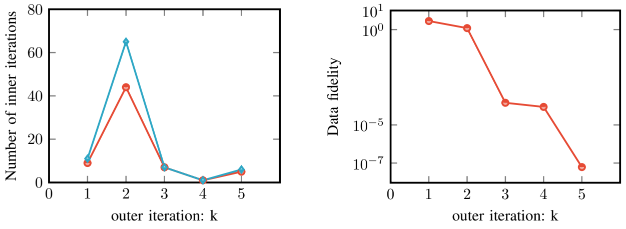
(ii) According to (51), we obtain

$$\|\boldsymbol{x}^\star - \boldsymbol{z}^\star\|_2^2 = \sum_{i \in \mathbb{S}(\boldsymbol{w}^\star)} \|\boldsymbol{x}^\star_{\mathbb{G}_i} - \boldsymbol{z}^\star_{\mathbb{G}_i}\|_2^2 = \sum_{i \in \mathbb{S}(\boldsymbol{w}^\star)} \|\boldsymbol{z}^\star_{\mathbb{G}_i}\|_2^2.$$

If the top-s strategy is applied, i.e. $\mathbb{S}^c(\boldsymbol{w}^\star) = \mathbb{A}_s(\boldsymbol{z}^\star)$, then for any feasible solution $\boldsymbol{x} \in \mathbb{C}_s$ we have $\|\boldsymbol{x} - \boldsymbol{z}^\star\|_2^2 \geq \sum_{i \in \mathbb{S}(\boldsymbol{w}^\star)} \|\boldsymbol{z}^\star_{\mathbb{G}_i}\|_2^2$, i.e., $\boldsymbol{x}^\star \in P_{\mathbb{C}_s}(\boldsymbol{x}^\star - \frac{1}{L}\nabla f(\boldsymbol{x}^\star))$.

If the best-s strategy is applied, i.e., $\mathbb{S}^c(\boldsymbol{w}^\star) = \mathbb{B}_s(\boldsymbol{z}^\star)$, then

$$\begin{aligned} &\mathcal{Y}(\mathrm{soft}_p(\boldsymbol{z}^\star_{\mathbb{G}_i}), 1) - \mathcal{Y}(\boldsymbol{z}^\star_{\mathbb{G}_i}, 0) \\ =&\mathcal{Y}(\mathrm{soft}_p(\boldsymbol{z}^\star_{\mathbb{G}_i}), 1) = \mathcal{Y}(\boldsymbol{0}, 1) = \|\boldsymbol{z}^\star_{\mathbb{G}_i}\|_2^2, \end{aligned}$$



(a) (b)

Fig. 8. Convergence behavior of HIWT-GSC along the homotopy path, the setting of data generation is $(1000, 4000, 1000, 50, 4, 0)$. (a) Number of inner iterations. The blue line shows the total number of iterations containing all line searches (failed and successful), and the red line gives the number of iterations containing only successful line searches. (b) Data fidelity measured by $\|\mathbf{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2$.

where $p \in \{1, 2\}$. At this time, the effect of the best-s strategy is the same as the top-s strategy. Then for any feasible solution $\boldsymbol{x}$ we have $\|\boldsymbol{x} - \boldsymbol{z}^\star\|_2^2 \geq \sum_{i \in \mathbb{S}(\boldsymbol{w}^\star)} \|\boldsymbol{z}^\star_{\mathbb{G}_i}\|_2^2$, i.e., $\boldsymbol{x}^\star \in P_{\mathbb{C}_s}(\boldsymbol{x}^\star - \frac{1}{L}\nabla f(\boldsymbol{x}^\star))$. ■

Appendix K
Empirical Study on the Number of Inner and Outer Iterations of the Algorithm

In this appendix, we conduct an empirical study on the number of inner and outer iterations of the HIWT-GSC algorithm through experiments on noiseless data.

In practice, the number of outer iterations can be set according to the needs of the application. For example, if we need λ to be a large enough value and do not want the λ to grow too fast, then the number of outer iterations can be set to a larger value. However, the final number of outer iterations depends on whether the termination conditions (see line 8 of the HIWT-GSC algorithm) can be satisfied in advance.

On the other hand, the number of inner iterations depends strongly on the accuracy we need for the solution. For each accepted solution, we use the inequality

$$\frac{\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|}{\|\boldsymbol{x}^k\|} < \epsilon$$

as a measure of accuracy. In line 5 of the HIWT-GSC algorithm, we gradually increase the accuracy requirement of IWT-GSC, i.e., we decrease ϵ_k as the iteration progresses.

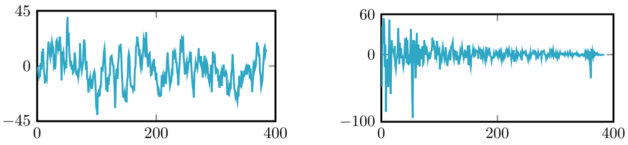
Next, we illustrate these with an example. In this test, the experimental settings are the same as in the main text (see page 7 for details). The data generation parameters were $(1000, 4000, 1000, 50, 4, 0)$, and the number of outer iterations was set to 10. We set $s_0 = s/4$ and exponentially increased s_k by $s_{k+1} = 2s_k$ until reaching the desired value s. We set $\lambda_0 = 0.1, \epsilon_0 = 10^{-2}$. And the default increasing factor of the sequence $\{\lambda_k\}$ was set as $\rho = 2$ and the decreasing factor of the sequence $\{\epsilon_k\}$ was set as $\varrho = 0.2$, i.e., $\lambda_{k+1} = 2\lambda_k, \epsilon_{k+1} = 0.2\epsilon_k$. The results are shown in Fig. 8.

As can be seen, the number of outer iterations performed is not 10, but 5, since the termination condition was satisfied in advance. We depict in Fig. 8a the number of inner iterations at each fixed λ and s. In the second outer iteration, since our algorithm uses a tighter termination tolerance than in the 1st iteration ($\epsilon_2 = 0.2\epsilon_1$), and the value of s_k has not yet reached the desired s, the number of inner iterations increases significantly. In the 3rd and 4th iterations, the value of s_k reaches the desired s and the IWT-GSC algorithm converges faster, which benefits from the warmstart strategy. It can be seen from Fig. 8b that, if the support set is correctly found (at the 3rd iteration), then the quality of the solution can be significantly improved, and in the final iteration, the quality of the solution can also be significantly improved by the debiasing step.

Appendix L
Application to electroencephalography signal reconstruction

In this appendix, the physiological signal reconstruction experiment is used to verify the performance of the proposed HIWT-GSC algorithm in the less sparse case.

By recording the electrical activity of the brain, electroencephalography (EEG) signals play an important role in the diagnosis of neurological diseases or disorders. This test was carried out on the EEGLab dataset [2]. The dataset contains 32 channels of EEG data, with each channel containing 80 epochs and each epoch containing 384 time points. Fig. 9a shows the EEG signals from the first epoch of the fifth channel in "eeglab_data.set" and Fig. 9b shows their DCT coefficients. It can be seen that EEG signals may not be sparse even in the transform domain, which can further complicate the reconstruction process.



(a) (b)

Fig. 9. EEG signals from the fifth epoch of the first channel in the EEGLAB Dataset, and their DCT coefficients. (a) A segment of EEG, (b) DCT coefficients

In this test, we apply group sparse optimization algorithms to reconstruct EEG signals. These algorithms use a group sparsity constraint to promote the sparsity of the EEG signals in a group-wise manner, thus allowing the reconstruction of EEG signals from a reduced set of electrodes.

Similar to [3], we can define a block partition with block size $d = 24$ in each channel, i.e., the signal structure in Fig. 10a can be viewed as

$$\boldsymbol{x} = (\underbrace{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_{24}}_{\boldsymbol{x}_{\mathbb{G}_1}^T},\ldots,\underbrace{\boldsymbol{x}_{361},\ldots,\boldsymbol{x}_{384}}_{\boldsymbol{x}_{\mathbb{G}_{16}}^T})^T.$$

We first linearly compressed $\boldsymbol{x}$ epoch by epoch with a sensing matrix and then expanded $\boldsymbol{x}$ in an inverse DCT basis $\boldsymbol{\Phi} = (\boldsymbol{\Phi}_1, \boldsymbol{\Phi}_2, \ldots, \boldsymbol{\Phi}_{384})$, i.e.,

$$\boldsymbol{b} = \mathbf{A}\boldsymbol{x} = \mathbf{A}\boldsymbol{\Phi}\boldsymbol{\alpha},$$

where $\boldsymbol{\alpha}$ is the coefficient vector and $\mathbf{A}$ is the same 192×384 sparse binary matrix as in [3].

In such a compressed sensing application, algorithm performance is usually measured in terms of the fidelity to the original signals. Therefore, we measured recovery quality using two metrics. One is the NMSE and the other is the Structural Similarity Index (SSIM) [4][5] for 1-D signals.

By setting the upper bound of the sparsity level ($\frac{s}{N}\%$) in HIWT-GSC to 10% and using the "best-s" strategy with $p = 2$, the average results of the respective algorithms on the whole data set are shown in TABLE IV. Since EEG signals are not sparse in both the time and transformed domains [6], many sparse recovery algorithms fail to achieve the desired recovery quality, such as GOMP and GPDASC. In comparison, GCD, SNSG and GSPGL1 perform better than GOMP and GPDASC in this test. From TABLE IV, it can be seen that our approach achieves better results than the other competing ones in such a difficult case. In addition, the performance of different algorithms varies at different group sizes. But in general, grouping exploits the continuity of the signal in the timing and provides better recovery quality than the case without grouping. It is worth noting that since both element and group sparsity are considered together in SNSG, it is not sensitive to group size in this test.

To visually detail the recovery performance of all the algorithms compared, we show in Fig. 10 the recovery quality of each algorithm for the first epoch of the first channel with group size $d = 24$. From the figure, it is easy to see that our method performs better than other competing ones, and its good reconstruction quality ensures subsequent signal analysis.

References

[1] H. Li and Z.-C. Lin, "Accelerated proximal gradient methods for nonconvex programming," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 379–387.

[2] A. Delorme and S. Makeig, "EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis," *J. Neurosci. Methods*, vol. 134, no. 1, pp. 9–21, Mar. 2004.

[3] Z. Zhang, T.-P. Jung, S. Makeig, and B. D. Rao, "Compressed sensing of EEG for wireless telemonitoring with low energy consumption and inexpensive hardware," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 1, pp. 221–224, Jan. 2013.

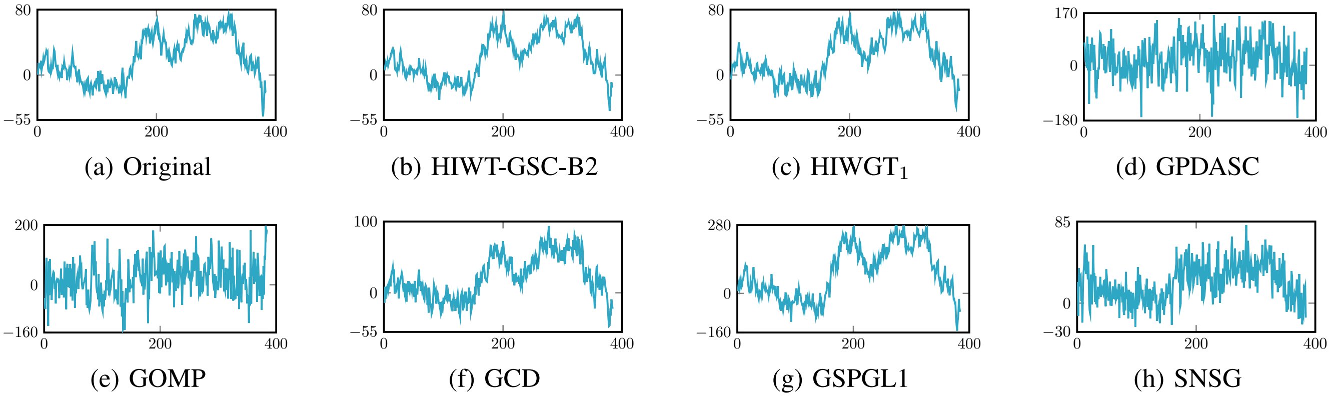
[4] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[5] Zhou Wang and A. Bovik, "Mean squared error: Love it or leave it? a new look at signal fidelity measures," *IEEE Signal Process. Mag.*, vol. 26, no. 1, pp. 98–117, Jan. 2009.

[6] Z.-L. Zhang, T.-P. Jung, S. Makeig, and B. D. Rao, "Compressed sensing of EEG for wireless telemonitoring with low energy consumption and inexpensive hardware," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 1, pp. 221–224, Jan. 2013.

TABLE IV
THE AVERAGE RECONSTRUCTION PERFORMANCE OF ONE EPOCH EEG SIGNAL IN ONE CHANNEL BY DIFFERENT METHODS WITH VARIED GROUP SIZE

Group size	Metrics	results of different algorithms						
		HIWT-GSC-B2	HIWGT$_1$	GPDASC	GOMP	GCD	GSPGL1	SNSG
1	NMSE	**0.1989**	0.3075	493.6406	0.3856	0.3774	7.4639	0.4611
	SSIM	**0.7013**	0.6333	0.4441	0.2999	0.5998	0.2034	0.4395
6	NMSE	**0.0976**	0.1671	19747.6877	136.2859	0.3254	7.1424	0.4424
	SSIM	**0.8134**	0.7618	0.1251	0.2999	0.6559	0.2228	0.4348
12	NMSE	**0.0904**	0.1528	596481.0727	1290.5444	0.3278	7.0243	0.4424
	SSIM	**0.8227**	0.7758	0.0797	0.2103	0.6587	0.2265	0.4348
16	NMSE	**0.0870**	0.1399	722780.7272	19545.6463	0.3014	6.9716	0.4424
	SSIM	**0.8233**	0.7875	0.0705	0.1820	0.6785	0.2283	0.4348
24	NMSE	**0.0904**	0.1502	91642.7825	2725.3443	0.3169	6.8609	0.4424
	SSIM	**0.8238**	0.7801	0.0553	0.1545	0.6654	0.2316	0.4348
32	NMSE	**0.0916**	0.1411	24699.7895	770.5234	0.2880	6.6968	0.4424
	SSIM	**0.8215**	0.7857	0.0533	0.1446	0.6841	0.2346	0.4348
48	NMSE	**0.1169**	0.1390	61084.1496	255.5278	0.2676	6.4823	0.4424
	SSIM	0.7864	**0.7885**	0.0620	0.1294	0.6980	0.2386	0.4348



(a) Original (b) HIWT-GSC-B2 (c) HIWGT$_1$ (d) GPDASC

(e) GOMP (f) GCD (g) GSPGL1 (h) SNSG

Fig. 10. Comparison of an original EEG epoch and the reconstructed results by different methods with a group size equal to 24. The horizontal axis: time points. The vertical axis: signal amplitude. (a) The first 384 time points of the first channel in the "eeglab_data.set". (b) The reconstructed segment by HIWT-GSC-B2, NMSE= 0.0228, SSIM= 0.8701. (c) The reconstructed segment by HIWGT$_1$, NMSE= 0.0378, SSIM= 0.8196. (d) The reconstructed segment by GPDASC, NMSE= 2.6379, SSIM= 0.0945. (e) The reconstructed segment by GOMP, NMSE= 2.9747, SSIM= 0.0724. (f) The reconstructed segment by GCD, NMSE= 0.0767, SSIM= 0.7883. (g) The reconstructed segment by GSPGL1, NMSE= 8.2876, SSIM= 0.2276. (h) The reconstructed segment by SNSG, NMSE= 0.3175, SSIM= 0.2625.