Tractability of Interpretability via Selection of Group-Sparse Models

Nirav Bhan
LIONS, EPFL, Switzerland

Luca Baldassarre
LIONS, EPFL, Switzerland

Volkan Cevher
LIONS, EPFL, Switzerland

***Abstract*—Group-based sparsity models are proven instrumental in linear regression problems for recovering signals from much fewer measurements than standard compressive sensing. A promise of these models is to lead to "interpretable" signals for which we identify its constituent groups, however we show that, in general, claims of correctly identifying the groups with convex relaxations would lead to polynomial time solution algorithms for an NP-hard problem. Instead, leveraging a graph-based understanding of group models, we describe group structures which enable correct model identification in polynomial time via dynamic programming. We also show that group structures that lead to totally unimodular constraints have tractable relaxations. Finally, we highlight the non-convexity of the Pareto frontier of group-sparse approximations and what it means for tractability.**

***Index Terms*—Signal Approximation, Structured Sparsity, Interpretability, Tractability, Dynamic Programming.**

I. Introduction

Recent extensions of compressive sensing move beyond the simple sparsity model to consider more sophisticated *structured* sparsity models, that allow to reduce the number of required measurements for perfect recovery in the noiseless case from $O(K \log N/K)$ down to $O(K)$ [1]–[4], where N is the ambient dimension and K is the sparsity level. The core intuition behind these achievements is that of customizing the geometry of the ensuing optimization problem to that of the class of signals we are interested in recovering.

Most of the structured sparsity models are based on groups of variables that should either be selected or discarded together [4]–[7]. These structures naturally arise in applications such as neuroimaging [8], [9], gene expression data [6], bioinformatics [10] and computer vision [3], [11]. While in signal processing we are mostly concerned with recovering a signal leveraging prior information, often in machine learning applications it is more important to discover the set of groups that constitute the signal support, in order to make inferences about possible underlying processes and causes. In fact, in many applications the groups have specific meanings and it is relevant to understand which groups are active and which are inactive. For example, in cancer research, the groups might represent genetic pathways that describe cellular processes. Identifying which processes lead to the development of a tumor can allow biologists to directly target certain groups of genes instead of others. Incorrect identification of the active/inactive groups can thus seriously affect the speed at which therapies are developed.

This work was supported in part by the European Commission under Grant MIRG-268398, ERC Future Proof, and SNF 200021-132548.

In this paper, we show that finding a group-sparse approximation of a signal, when the groups can overlap arbitrarily, is in general an intractable problem. Our results stem from the reformulation of the signal approximation problem as the Weighted Maximum Coverage (WMC) integer optimization problem, which is a well-known combinatorial problem that is NP-hard. By leveraging a graph-based representation of the groups, we present characterizations of group structures that lead to tractable cases. For loopless pairwise overlapping groups, we develop a dynamical programming algorithm that solves the integer problem in polynomial time. We also show that, if the group structure satisfies a totally unimodular constraint, we can relax the integer problem into a binary linear program that can be solved in polynomial time.

We then extend our model to incorporate sparsity constraints, leading to a generalization of the WMC problem. We show that the generalized model can be solved in polynomial time by a dynamic program for loopless pairwise overlapping groups and hierarchical group structures. Furthermore, by relaxing the constraints on the number of groups and on the sparsity, we obtain a binary linear program that can be solved in polynomial time if the constraints imposed by the group structure are totally unimodular.

Finally, we illustrate the non-convexity of the Pareto frontier of group-sparse approximations in a simple signal approximation problem using hierarchical constraints. Totally unimodular discrete relaxations and convex relaxations can only obtain solutions on the convex hull of the Pareto frontier, potentially leading to erroneous interpretations of the signal. Indeed, if the relaxations were able to yield all possible solutions, they would provide polynomial-time algorithms for solving an NP-hard problem. Given space limitations, proofs of our results can be found in [14].

II. Basic Definitions

Let $x \in \mathbb{R}^N$ be a vector and $\mathcal{N} = \{1, \dots, N\}$ be the ground set of its indices. We use $|\mathcal{S}|$ to denote the cardinality of an index set $\mathcal{S}$. We use $\mathbb{B}^N$ to represent the space of N-dimensional binary vectors and define $\iota : \mathbb{R}^N \to \mathbb{B}^N$ to be the indicator function of the nonzero components of a vector in $\mathbb{R}^N$, i.e. $\iota(x)_i = 1$ if $x_i \neq 0$ and $\iota(x)_i = 0$ otherwise. We let $\mathbf{1}_N$ to be the N-dimensional vector of all ones and $\mathbf{I}_N$ the

978-1-4799-0446-4/13/$31.00 ©2013 IEEE

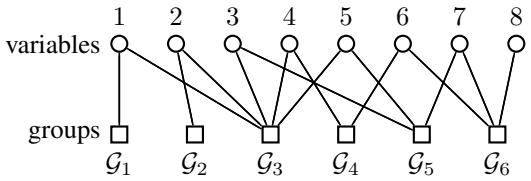

Fig. 1. Example of bipartite graph between variables and groups induced by the group structure $\mathfrak{G}^1$, see text for details.

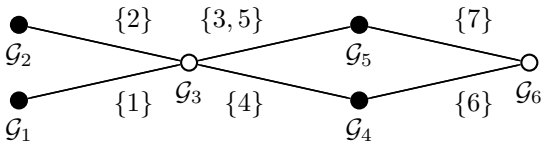

Fig. 2. Bipartite group graph with loops induced by the group structure $\mathfrak{G}^1$, where on each edge we report the elements of the intersection.

$N \times N$ identity matrix. The support of x is defined by the set-valued function $\operatorname{supp}(x) = \{i \in \mathcal{N} : x_i \neq 0\}$.

Definition II.1. *A* **group structure** $\mathfrak{G} = \{\mathcal{G}_1, \ldots, \mathcal{G}_M\}$ *is a collection of index sets, named* groups*, with* $\mathcal{G}_j \subseteq \mathcal{N}$ *and* $|\mathcal{G}_j| = g_j$ *for* $1 \leq j \leq M$ *and* $\bigcup_{\mathcal{G}\in\mathfrak{G}} \mathcal{G} = \mathcal{N}$.

We can represent a group structure $\mathfrak{G}$ as a bipartite graph, where on one side we have the N variables nodes and on the other the M group nodes. An edge connects a variable node i to a group node j if $i \in \mathcal{G}_j$. The incidence matrix $\mathbf{A}^{\mathfrak{G}} \in \mathbb{B}^{N\times M}$ of the bipartite graph encodes the group structure: $\mathbf{A}^{\mathfrak{G}}_{ij} = 1$ if $i \in \mathcal{G}_j$ and $\mathbf{A}^{\mathfrak{G}}_{ij} = 0$ otherwise.

Another useful representation of a group structure is via a *group graph* (V, E) where the nodes V are the groups $\mathcal{G} \in \mathfrak{G}$ and the edge set E contains e_{ij} if $\mathcal{G}_i \cap \mathcal{G}_j \neq \emptyset$, that is an edge connects two groups that *overlap*. A sequence of connected nodes $v_1, v_2, \ldots, v_n$, is a *loop* if $v_1 = v_n$.

In order to illustrate these concepts, consider the group structure $\mathfrak{G}^1$ defined by the following groups, $\mathcal{G}_1 = \{1\}$, $\mathcal{G}_2 = \{2\}$, $\mathcal{G}_3 = \{1, 2, 3, 4, 5\}$, $\mathcal{G}_4 = \{4, 6\}$, $\mathcal{G}_5 = \{3, 5, 7\}$ and $\mathcal{G}_6 = \{6, 7, 8\}$. $\mathfrak{G}^1$ can be represented by the variables-groups bipartite graph of Fig. 1 or by the group graph of Fig. 2, which is bipartite and contains loops.

An important group structure is given by *loopless pairwise overlapping groups*. This group structure consists of groups such that each element of the ground set occurs in at most two groups and the induced graph does not contain loops.

We anchor our analysis on covering arguments.

Definition II.2. *A* **group cover** $S(x)$ *for a signal* $x \in \mathbb{R}^N$ *is a collection of groups such that* $\operatorname{supp}(x) \subseteq \bigcup_{\mathcal{G}\in S(x)} \mathcal{G}$*:*

$$S(x) = \{\mathcal{G}_j \in \mathfrak{G} : \omega \in \mathbb{B}^M,\ \omega_j = 1,\ \mathbf{A}^{\mathfrak{G}}\omega \geq \iota(x)\}\ .$$

The binary vector ω indicates which groups are active and the constraint $\mathbf{A}^{\mathfrak{G}}\omega \geq \iota(x)$ makes sure that for every non-zero component of x at least a group that covers it is active. We also say that $S(x)$ *covers* x since $\operatorname{supp}(x) \subseteq \bigcup_{\mathcal{G}\in S(x)} \mathcal{G}$. Note that the group cover is often not unique and $S(x) = \mathfrak{G}$ is a group cover for any signal x. This observation leads us to consider more restrictive definitions of group cover.

Definition II.3. *A* G**-group cover** $S^G(x) \subseteq \mathfrak{G}$ *is a group cover for* x *with at most* G *elements.*

It is not guaranteed that a G-group cover always exists for any value of G. Finding the smallest G-group cover lead to the following definitions.

Definition II.4. *The* **group** ℓ_0**-"norm"** *is defined as*

$$\|x\|_{\mathfrak{G},0} := \min_{\omega\in\mathbb{B}^M} \left\{ \sum_{j=1}^{M} \omega_j : \mathbf{A}^{\mathfrak{G}}\omega \geq \iota(x) \right\}\ . \tag{1}$$

Definition II.5. *A* **minimal group cover** *for a signal* $x \in \mathbb{R}^N$ *is defined as* $\mathcal{M}(x) = \{\mathcal{G}_j \in \mathfrak{G} \mid \hat{\omega}(x)_j = 1\}$*, where* $\hat{\omega}$ *is a minimizer for* (1)*.*

A *minimal group cover* $\mathcal{M}(x)$ is a group cover for x with minimal cardinality. Note that there exist pathological cases where for the same group ℓ_0-"norm", we have different minimal group cover models.

Definition II.6. *A signal* x *is* G**-group sparse** *with respect to a group structure* $\mathfrak{G}$ *if* $\|x\|_{\mathfrak{G},0} \leq G$*.*

III. Tractability of interpretations

In this paper, we address the problem of finding group-based interpretations of a signal x. We first define the G-group sparse approximation $\hat{x}$ and then show that it it can be easily obtained from its G-group cover $S^G(\hat{x})$. In many applications, the group cover S^G of an approximation is more important than the actual coefficients, because it allows to "interpret" the solution: by indicating the active groups of variables, we can take actions, such as targeting specific pathways for cancer diagnosis or treatment.

Problem 1 (Signal approximation)**.** *Given a signal* $x \in \mathbb{R}^N$*, a best* G*-group sparse approximation* $\hat{x}$ *is*

$$\hat{x} \in \operatorname*{argmin}_{z\in\mathbb{R}^N} \left\{ \|x - z\|_2^2 : \|z\|_{\mathfrak{G},0} \leq G \right\} \tag{2}$$

Problem 2 (Model selection)**.** *Given a signal* $x \in \mathbb{R}^N$*, a* G*-group cover model for its* G*-group sparse approximation is*

$$S^G(\hat{x}) \in \operatorname*{argmax}_{\substack{S\subseteq\mathfrak{G} \\ |S|\leq G}} \left\{ \sum_{i\in I} x_i^2 : I = \bigcup_{\mathcal{G}\in S} \mathcal{G} \right\} \tag{3}$$

Given $S^G(\hat{x})$, we can obtain $\hat{x}$ as $\hat{x}_I = x_I$ and $\hat{x}_{I^c} = 0$, where $I = \bigcup_{\mathcal{G}\in S^G(\hat{x})} \mathcal{G}$ and $I^c = \mathcal{N} \setminus I$.

The following reformulation of Problem 2 as a binary problem allows us to characterize its tractability.

Lemma 1. *Given* $x \in \mathbb{R}^N$ *and a group structure* $\mathfrak{G}$*, we have*

that $S^G(\hat{x}) = \{\mathcal{G}_j \in \mathfrak{G} \mid \omega_j^G = 1\}$*, where* (ω^G, y^G) *solves*

$$\max_{\omega \in \mathbb{B}^M,\ y \in \mathbb{B}^N} \left\{ \sum_{i=1}^{N} y_i x_i^2 : \mathbf{A}^{\mathfrak{G}} \omega \geq y, \sum_{j=1}^{M} \omega_j \leq G \right\} \quad (4)$$

Problem (4) can produce all instanced of the Weighted Maximum Coverage problem (WMC), which is known to be NP-hard. Given Lemma 1, the tractability of (3) is a consequence of the hardness of (4).

Proposition III.1. *The model selection problem* (3) *is in general NP-hard.*

It is possible to approximate the solution of (4) using the greedy WMC algorithm [12]. At each iteration, the algorithm selects the group that covers new variables with maximum combined weight until G groups have been selected.

Our main result (see Appendix) is an algorithm for solving (4) exactly for loopless pairwise overlapping groups structures.

Proposition III.2. *Given a loopless pairwise overlapping group structure* $\mathfrak{G}$*, there exists a polynomial time dynamic programming algorithm that solves* (4)*.*

IV. Discrete relaxations

If we relax the constraint on the number of groups in (4) into a regularization term with parameter $\lambda > 0$, we obtain the following binary linear program

$$(\omega^\lambda, y^\lambda) \in \operatorname*{argmax}_{\omega \in \mathbb{B}^M, y \in \mathbb{B}^N} \left\{ \mathbf{w}^\top \mathbf{u} : \mathbf{u}^\top = [y^\top\ \omega^\top],\ \mathbf{C}\mathbf{u} \leq 0 \right\} \quad (5)$$

where $\mathbf{w}^\top = [x_1^2, \ldots, x_N^2, -\lambda \mathbf{1}_M^\top]$ and $\mathbf{C} = [\mathbf{I}_N,\ -\mathbf{A}^{\mathfrak{G}}]$.

In general, (5) is NP-hard to solve exactly, however, it is well known [13] that if the constraint matrix $\mathbf{C}$ is Totally Unimodular (TU) then there exist polynomial-time algorithms for solving it. Due to its structure, $\mathbf{C}$ is TU if $\mathbf{A}^{\mathfrak{G}}$ is TU [13].

Group structures that can be represented by a bipartite graph and admit only pairwise overlaps, such as the one in Fig.2 lead to constraint matrices $\mathbf{A}$ that are TU [13].

Lemma 2. *Loopless pairwise overlapping groups lead to totally unimodular constraints.*

Even though for this group structure we can use the DP algorithm of Prop.III.2, for very large problems it may be computationally faster to solve the binary linear program. The next proposition establishes when the regularized solution coincides with the solution of (4).

Proposition IV.1. *If the value of the regularization parameter* λ *is such that the solution* $(\omega^\lambda, y^\lambda)$ *of* (5) *satisfies* $\sum_j \omega_j^\lambda = G$*, then* $(\omega^\lambda, y^\lambda)$ *is also a solution for* (4)*.*

However, as we show in Sect.VIII, given a value of G it is not always possible to find a value of λ such that the solution of (5) is also a solution for (4). More specifically, we can only obtain the solutions that lie on the convex hull of the Pareto frontier of (4), which is the set of points $\{G, (f(G))\}_{G=1}^M$, where $f(G) = \sum_{i=1}^N y_i^G x_i^2$ [14].

V. Convex relaxations

For tractability and analysis, a convex proxies of the group ℓ_0-norm have been introduced [15] for finding a group-sparse approximation of a signal. Given a group structure $\mathfrak{G}$, an example generalization is defined as

$$\|x\|_{\mathfrak{G},\{1,p\}} := \inf_{\substack{\mathbf{v}^1, \ldots, \mathbf{v}^M \in \mathbb{R}^N \\ \forall j,\, \mathrm{supp}(\mathbf{v}^j) = \mathcal{G}_j}} \left\{ \sum_{j=1}^{M} d_j \|\mathbf{v}^j\|_p : \sum_{j=1}^{M} \mathbf{v}^j = x \right\} \quad (6)$$

where $\|x\|_p = \left(\sum_{i=1}^N x_i^p\right)^{1/p}$ and d_j are positive weights that can be designed to favor certain groups over others [6]. This norm can be seen a weighted generalization of the atomic norm described in [4], where they characterize its properties for signal recovery, but not for model selection. We can use (6) to find a group-sparse approximation

$$\hat{x} = \operatorname*{argmin}_{z \in \mathbb{R}^N} \left\{ \|x - z\|_2^2 : \|z\|_{\mathfrak{G},\{1,p\}} \leq \lambda \right\} \quad (7)$$

where $\lambda > 0$ controls the trade-off between approximation accuracy and group-sparsity. However, solving (7) does not yield a group-support for $\hat{x}$: even though we can recover one through the decomposition $\{\mathbf{v}^j\}$ used to compute $\|\hat{x}\|_{\mathfrak{G},\{1,p\}}$, it may not be unique as observed in [6] for $p = 2$. In order to characterize the group-support for x induced by (6), [6] define two group-supports for $p = 2$. The *strong group-support* $\check{\mathcal{S}}(x)$ contains the groups that constitute the supports of each decomposition used for computing (6). The *weak group-support* $\mathcal{S}(x)$ is defined using a dual-characterization of the group norm (6). If $\check{\mathcal{S}}(x) = \mathcal{S}(x)$, the group-support is uniquely defined. However, [6] observed that for some group structures and signals x, even when $\check{\mathcal{S}}(x) = \mathcal{S}(x)$, the group-support does not capture the minimal group-cover of x.

VI. Discrete vs. Convex Interpretability

The following stylized example illustrates situations that can potentially be encountered in practice. In these cases, the group-support obtained by the convex relaxation will not coincide with the discrete definition of group-cover, while the dynamical programming algorithm of Prop. III.2 is able to recover the correct group-cover.

Let $\mathcal{N} = \{1, \ldots, 11\}$ and let $\mathfrak{G} = \{\mathcal{G}_1 = \{1, \ldots, 5\},\ \mathcal{G}_2 = \{4, \ldots, 8\},\ \mathcal{G}_3 = \{7, \ldots, 11\}\}$ be a loopless pairwise overlapping groups structure with 3 groups. Consider the 2-group sparse signal $x = [0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0]^\top$, with minimal group-cover $\mathcal{M}(x) = \{\mathcal{G}_1, \mathcal{G}_3\}$.

The dynamic program of Prop.III.2, with group budget $G = 2$, correctly identifies the groups $\mathcal{G}_1$ and $\mathcal{G}_3$. The TU linear program (5), with $0 < \lambda \leq 2$, also yields the correct group-cover. Conversely, the decomposition obtained via (6) with unitary weights is unique, but is not group sparse. In fact, we have $\mathcal{S}(x) = \check{\mathcal{S}}(x) = \mathfrak{G}$. We can only obtain the correct

group-cover if we use the weights $[1\ d\ 1]$ with $d > \frac{2}{\sqrt{3}}$, that is knowing beforehand that $\mathcal{G}_2$ might be irrelevant.

Remark 1. *Indeed, if the convex relaxation always recovered the correct minimal group-cover, it would be possible to solve the discrete NP-hard problem in polynomial time.*

VII. Generalizations

We can generalize (4) by introducing a sparsity constraint K and allowing to individually select variables within a group. The generalized integer problem then becomes

$$\max_{\omega\in\mathbb{B}^M,\ y\in\mathbb{B}^N} \left\{ \sum_{i=1}^{N} y_i x_i^2 : \mathbf{A}^{\mathfrak{G}}\omega \geq y, \sum_{i=1}^{N} y_i \leq K, \sum_{j=1}^{M} \omega_j \leq G \right\}. \tag{8}$$

We can modify the dynamic programming algorithm of Prop.III.2, in order to solve (8) (see proof in the Appendix).

Proposition VII.1. *Given a loopless pairwise overlapping groups structure $\mathfrak{G}$, there exists a polynomial time dynamic programming algorithm that solves* (8).

The generalized model allows to deal with hierarchical structures, such as regular trees, frequently encountered in image processing (e.g. denoising using wavelet trees). In such cases, we often require to find K-sparse approximations such that the selected variables form a rooted connected sub-tree of the original tree. This type of constraint can be represented by a group structure, where for each node in the tree we define a group consisting of that node and all its ancestors. When a group is selected, we also require that all its elements are selected as well and impose a sparsity constraint K, but not a group constraint G. For this particular problem, for which convex approximations have been proposed [16], we present an exact dynamic program that runs in polynomial time.

Proposition VII.2. *Given a hierarchical group structure $\mathfrak{G}$, there exists a polynomial time dynamic programming algorithm that solves* (8).

By relaxing both the group budget and the sparsity budget in (8) into regularization terms, we obtain a binary linear program, which can be solved in polynomial time if the constraint matrix is TU. We already established TU for loopless pairwise overlapping groups. TU also holds for hierarchical groups.

Proposition VII.3. *Hierarchical group structures lead to to totally unimodular constraints.*

VIII. Pareto Frontier Example

The purpose of this numerical simulation is to illustrate the limitations of relaxations for correctly estimating the G-group cover of an approximation. We consider the problem of finding a K-sparse approximation of a signal imposing hierarchical constraints. We generate a piecewise constant signal of length $N = 64$, to which we apply the Haar wavelet transformation,

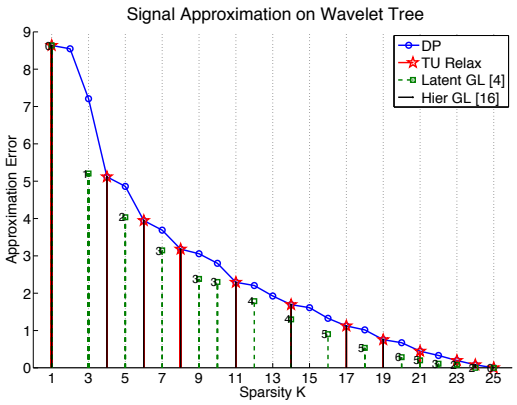

Fig. 3. Signal approximation on a binary tree. The original signal is 25-sparse and satisfies hierarchical constraints. The numbers next to the Latent Group Lasso solutions indicate the number of constraint violations.

yielding a 25-sparse vector of coefficients x that satisfies hierarchical constraints on a binary tree of depth 5.

We compare the DP to the regularized TU approach and two convex relaxations that use group-based norms. The first [4] uses the latent group lasso penalty (7) with groups defined as all father-child relations in the tree. This formulation will not enforce all hierarchical constraints to be satisfied, but will only 'favor' them. Therefore, we also report the number of hierarchical constraint violations. The second [16] considers a hierarchy of groups where $\mathcal{G}_j$ contains node j and all its descendants. Hierarchical constraints are enforced by the group lasso penalty $\Omega_{GL}(x) = \sum_{\mathcal{G}\in\mathfrak{G}} \|x_{\mathcal{G}}\|_2$, where $x_{\mathcal{G}}$ is the restriction of x to $\mathcal{G}$. Once we determine the support of the solution, we assign to the components in the support the values of the corresponding components of the original signal.

In Fig.3, we show the approximation error $\|x - \hat{x}\|_2^2$ as a function of the solution sparsity K for the methods. The values of the DP solutions form a discrete and non-convex Pareto frontier of the optimization problem controlled by the parameter K. With the TU relaxation we are only able to observe the solutions that lie in the boundary of the convex hull of the Pareto frontier, while the Hierarchical Group Lasso also misses $K = 21$ and $K = 23$. The Latent Group Lasso approach achieves more levels of sparsity (but still missing the solutions for $K = 2, 13$ and 15), although at the price of violating some of the hierarchical constraints. These observations lead us to conclude that, in some cases, relaxations of the original discrete problem might lead to erroneous identification of the underlying sparsity model.

IX. Conclusions

We show that finding a group-based interpretation of a signal is an integer optimization problem, which is in general NP-hard. Leveraging a graph representation, we characterize group structures for which a dynamical programming algorithm yields a solution in polynomial time. We also present discrete relaxations that lead to binary linear programs with totally

unimodular constraints, that can be solved in polynomial time.

Our examples and numerical simulations show the deficiencies of relaxations and especially of convex methods that rely on atomic norm minimization. In fact, these methods can only recover group-covers that lie in the convex hull of the Pareto frontier determined by the solutions of the original integer problem for different values of the group budget G (and sparsity budget K for the general case). This, in turn, implies that convex and non-convex relaxations might lead to erroneous group-based interpretations of signals.

APPENDIX
DYNAMICAL PROGRAMMING FOR SOLVING (8)

Here, we give a sketch of the proof of Prop.VII.2. Full details can be found in [14].

Proof: The proof consists in describing the algorithm and showing it is polynomial time. The algorithm gradually explores every node in the group-graph, storing the optimal solution among the visited nodes and it is defined by two rules: the **Node Picking Rule** controls how the graph must be explored in order to minimize the number of values to store; the **Value Update Rule** describes how the stored values are updated when a new node is considered. Due to the looplessness constraint, the graph can be represented as a tree. Choose an arbitrary node and call it the root node.

Suppose we have explored m out of the total M nodes. We store the best possible value that can be obtained by choosing at most g nodes and at most k elements from the currently explored set of nodes, for $1 \leq g \leq G$ and $1 \leq k \leq K$. We expand the set of explored nodes one new node at a time and recompute the set of stored values. Define a **boundary node** as an explored node adjacent to an unexplored node. We must store all values separately for each possible selection of boundary nodes. The value update rule is divided into 3 cases.

1) *New node is rejected*. All optimal values for k and g remain the same. The added node is treated as a new boundary node and the values correspond to rejecting it.
2) *New node is accepted, no overlap with any explored node*. Since the new node is selected, we can choose at most $g-1$ explored nodes. The new optimal value for each g and k is given by taking the maximum over $1 \leq \ell \leq k$ of the sum of the optimal value for choosing best ℓ elements from the new node and the optimal value for choosing $k-\ell$ elements from $g-1$ explored nodes.
3) *New node is accepted, overlaps with some explored nodes*. The update rule is the same as for case 2, but the elements in the region of overlap between the new node and the selected explored nodes must not be considered as being part of the new node.

After these steps, the number of stored values will be (at most) doubled. We can reduce them: for each boundary node which has fallen into the interior of the explored nodes, we combine the optimal values for it being picked or unpicked, into a single value by taking the larger of the 2 values.

Let B be the maximum number of boundary nodes encountered by the algorithm, then the number of steps is bounded by $O(2^B K^2 G M)$. We now give an algorithm to explore the graph so that B is logarithmic in M, establishing polynomial complexity.

We first order all rooted subtrees with an index called the D-value, so that $D_1 \geq \cdots \geq D_R$ for subtrees $T_1, T_2, \ldots, T_R$. We then pick the subtrees in the order $\{T_1, \text{root}, T_2, \ldots, T_R\}$ and recurse until the explored subtree has only one node.

The procedure for computing D-values is also recursive. Let the subtrees at a node Q have values $D_1 \geq \ldots \geq D_R$. Then, $D(Q) = \max(D_1, D_2 + 1)$. In case there is no 2nd subtree, $D_2 = 0$. If the graph has only one node, $D = 1$.

The following theorem establishes polynomial complexity of the algorithm (see [14] for the proof).

Theorem 1. *The maximum number of boundary nodes at any step of the algorithm is logarithmic in the number of nodes, i.e.* $B = O(\log M)$.

■

REFERENCES

[1] Y. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *Information Theory, IEEE Transactions on*, vol. 55, no. 11, pp. 5302–5316, 2009.

[2] T. Blumensath and M. Davies, "Sampling theorems for signals from the union of finite-dimensional linear subspaces," *Information Theory, IEEE Transactions on*, vol. 55, no. 4, pp. 1872–1882, 2009.

[3] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde, "Model-based compressive sensing," *Information Theory, IEEE Transactions on*, vol. 56, no. 4, pp. 1982–2001, 2010.

[4] N. Rao, B. Recht, and R. Nowak, "Signal recovery in unions of subspaces with applications to compressive imaging," *arXiv preprint arXiv:1209.3079*, 2012.

[5] R. Jenatton, J.-Y. Audibert, and F. Bach, "Structured variable selection with sparsity-inducing norms," *Journal of Machine Learning Research*, vol. 12, pp. 2777–2824, 2011.

[6] G. Obozinski, L. Jacob, and J. Vert, "Group lasso with overlaps: The latent group lasso approach," *arXiv preprint arXiv:1110.0413*, 2011.

[7] N. Rao, R. Nowak, S. Wright, and N. Kingsbury, "Convex approaches to model wavelet sparsity patterns," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*, 2011, pp. 1917–1920.

[8] A. Gramfort and M. Kowalski, "Improving m/eeg source localization-with an inter-condition sparse prior," in *IEEE International Symposium on Biomedical Imaging*, 2009.

[9] R. Jenatton, A. Gramfort, V. Michel, G. Obozinski, F. Bach, and B. Thirion, "Multi-scale mining of fmri data with hierarchical structured sparsity," in *Pattern Recognition in NeuroImaging (PRNI)*, 2011.

[10] F. Rapaport, E. Barillot, and J. Vert, "Classification of arraycgh data using fused svm," *Bioinformatics*, vol. 24, no. 13, pp. i375–i382, 2008.

[11] V. Cevher, C. Hegde, M. Duarte, and R. Baraniuk, "Sparse signal recovery using markov random fields," in *NIPS*, 2009.

[12] G. Nemhauser, L. Wolsey, and M. Fisher, "An analysis of approximations for maximizing submodular set functionsi," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[13] L. Wolsey and G. Nemhauser, *Integer and Combinatorial Optimization*. Wiley, 1999.

[14] L. Baldassarre, N. Bhan, V. Cevher, and A. Kyrillidis, "Group-sparse model selection: Hardness and relaxations," *arXiv preprint arXiv:1303.3207*, 2013.

[15] L. Jacob, G. Obozinski, and J. Vert, "Group lasso with overlap and graph lasso," in *International Conference on Machine Learning*, 2009.

[16] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach, "Proximal methods for hierarchical sparse coding," *Journal of Machine Learning Reasearch*, vol. 12, pp. 2297–2334, 2011.