Conjugate Gradient Iterative Hard Thresholding for Structured Sparsity

JEFFREY D. BLANCHARD
Department of Mathematics and Statistics, Grinnell College, Grinnell, IA 50112 USA

Dedicated with deep gratitude to Guido Leopold Weiss, 1928–2021 [1]

ABSTRACT Greedy sparse recovery algorithms are studied in the structured sparsity (sparsity in levels) framework. Recovery guarantees are provided for Normalized Iterative Hard Thresholding and Conjugate Gradient Iterative Hard Thresholding in the form of restricted isometry properties for sparsity in levels. Empirical results indicate that CGIHT is comparable to CoSaMP in recovery capability in the structured setting, while maintaining the computational complexity of NIHT. While exploiting structured sparsity improves recovery performance, pessimistic theoretical guarantees mask when practitioners should use these algorithms; the empirical results offer guidance for using the original greedy algorithms over CGIHT in Levels.

INDEX TERMS Compressed sensing, conjugate gradient iterative hard thresholding, greedy algorithms, model-based compressed sensing, structured sparsity, sparsity in levels, sparse approximation.

I. INTRODUCTION

A. COMPRESSED SENSING AND SPARSE APPROXIMATION

In the simplest noise-free settings of compressed sensing [2], [3] and sparse approximation [4], [5], we start with only the measurements, \( y \in \mathbb{R}^m \), and knowledge of the measurement operator, \( A \in \mathbb{R}^{m \times n} \). Given no other information, but assuming the measured vector is sparse, we seek the sparsest vector \( x \in \mathbb{R}^n \) which could produce the measurements \( y \). In other words, the compressed sensing problem can be stated as the combinatorial optimization

\[
x = \operatorname{arg\ min}_{z \in \mathbb{R}^n} \|z\|_0 \text{ subject to } y = Az.
\]

(1)

where \( \|z\|_0 \) counts the number of nonzero entries in \( z \). The closely related sparse approximation problem assumes the measured vector is well approximated by a sparse vector, and therefore seeks the best approximation of the measurements when the sensed vector is given a fixed sparsity allowance, namely

\[
x = \operatorname{arg\ min}_{z \in \mathbb{R}^n} \|y - Az\|_2 \text{ subject to } \|z\|_0 \leq k.
\]

(2)

Although both problems are NP-hard in general [6], much work has been done to find tractable alternatives to (1) and (2). Under the right conditions [7], [8], [9], [10], one can obtain the solution to (1) from its convex relaxation,

\[
x = \operatorname{arg\ min}_{z \in \mathbb{R}^n} \|z\|_1 \text{ subject to } y = Az.
\]

(3)

Alternatively, many iterative algorithms have been designed to directly conquer (2) [11], [12], [13], [14], [15], [16], [17], [18], [19], [20].

One important tool in the analysis of compressed sensing and sparse approximation is the ubiquitous restricted isometry property [9]. A measurement matrix \( A \) has the asymmetric restricted isometry constants \( L_{ck}, U_{ck} \) when these are the smallest constants so that

\[
(1 - L_{ck})\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + U_{ck})\|x\|_2^2.
\]

(4)

The large majority of recovery guarantees come in the form of some restricted isometry property (RIP) where the matrix \( A \) must have restricted isometry constants (RIC) satisfying a particular bound. For example, a necessary condition for \( A \) to be injective on the set of all \( k \)-sparse vectors, and therefore ensure a unique solution to (1) whenever \( x \) is \( k \)-sparse, is the restricted isometry property \( L_{2,k} < 1 \).
B. OUTLINE AND CONTRIBUTIONS
The remainder of this introduction briefly introduces the main contributions of the article and some notation used throughout. In Section II, we first recall the general advances in models of structure for compressed sensing (Section II-A). In Section II-B we describe three specific models and detail how they are essentially equivalent. Greedy algorithms designed to exploit the sparsity in levels model are described in Section II-C; in particular we introduce Conjugate Gradient Iterative Hard Thresholding for sparsity in levels.

The main results of the paper are presented in Section III. In this section, Normalized Iterative Hard Thresholding (NIHT) and Conjugate Gradient Iterative Hard Thresholding (CGIHT) are analyzed using asymmetric restricted isometry constants (aRICs). Theorems 1 and 2 provide guarantees for uniform recovery of signals with the sparsity in levels structure for both NIHT and CGIHT when the measurement operator has appropriately bounded aRICs.

Section IV presents several empirical investigations to enhance the theoretical results from Section III. Experiments indicate that the structure exploiting algorithms are bolstered not just by signals having significant structure, but also by the quality of structure information given to the algorithms. Standard and sparsity in levels adapted versions of the algorithms are compared when given incorrect, minimal, partial, or exact structure information on signals with varying degrees of sparsity in levels structure. For signals with a highly structured, dyadic sparsity pattern, the algorithms are compared and recommendations made on which algorithms are likely to have the most preferable average case behavior.

The proofs of the main results from Section III are provided in Appendix A. These proofs are both an extension of the proofs to the sparsity in levels setting and an improvement on the required bounds on the aRICs of the measurement operator in order to guarantee uniform recovery.

C. NOTATION
For the remainder, index sets will be denoted by a capital letter, such as $Q, S, T$, while most Greek and script letters will indicate other types of sets. Standard lower case letters will indicate integers, for example $c, k, m, n, r$. Vectors will be written in bold, $x, y, w, e$, with some special bold face letters $k = (k_1, \ldots, k_r), n = (n_1, \ldots, n_q)$ representing ordered tuples of integers. When a vector is partitioned into smaller portions of the vector, these portions are indicated with a subscript, e.g. $x = [x_1|\ldots|x_\tau]$. Superscript indices distinguish vectors from one another so that $x^1$ and $x^2$ are two different vectors, often with the superscript identifying the vector associated with an iteration of an algorithm. Matrices will be given a capital letter, such as $A$, and will be clear from context that this is not a set. For a vector $w \in \mathbb{R}^n$ and an index set $T$, the hard threshold of $w$ onto $T$ is defined by

$$ (w)_T = \begin{cases} w_i & i \in T \\ 0 & i \not\in T \end{cases} $$

For a matrix $A$, the restriction of $A$ to the columns indexed by $T$ will be denoted $AT$. Also clear from context, $|x|$ is the absolute value of the scalar $x$, $|w|$ is the vector where the absolute value is taken componentwise, and $|Q|$ is the number of elements in a discrete set $Q$.

II. STRUCTURED SPARSITY
In many applications, we know more than just the measurements $y$ and the operator $A$; we also know some structure to the information. For example, we may know that $x = \Psi s$ is the set of wavelet coefficients for the signal $s$ and therefore understand a dyadic structure of the important information in $x$. This additional structure information could be exploited to improve the reconstruction process.

A. UNIONS OF SUBSPACES
A general model of sparsity was put forward in [4] where the notion of sparsity was presented as having a sparse representation in a union of subspaces. Within the compressed sensing framework, the union of subspaces model was formalized and advanced in a large body of work including [21], [22], [23], [24].

Definition 1 (Union of Subspaces Model): For each $i \in \{1, \ldots, r\}$ let $\chi_i \subset \mathbb{R}^n$ be a $k$-dimensional subspace. The structured sparsity model $M_k$ is the union of $k$-dimensional subspaces

$$ M_k = \bigcup_{i=1}^r \chi_i. $$

In [21], Lu and Do provide several examples of such signal models including streams of Diracs in finite rate of innovation, piecewise polynomials, sparse representations in a dictionary, overlapping echoes, and signals with unknown spectral support. The authors study when a sensing operation is invertible and stable by analyzing a Gram matrix associated with the sampling and reconstruction schemes. The stability of the operators on a union of subspaces model is detailed in language equivalent to an asymmetric restricted isometry property.

In [22], Blumensath and Davies give an explicit definition for restricted isometry constants on a finite union of subspaces. They employ these RICs to describe both the injectivity of a linear mapping and a Lipschitz continuity condition (stability) on unions of $k$-dimensional subspaces. Importantly, [22] establishes that, with high probability, matrices with subgussian entries have bounded RICs for the union of subspaces models.

In [23], the iterative greedy algorithms Iterative Hard Thresholding (IHT) [13], [14] and Compressive Sampling Matching Pursuit (CoSaMP) [16] were analyzed by Baron, Blumensath and Davies and for the union of subspaces model. In this very general setting, recovery guarantees are established for both algorithms in terms of a restricted isometry property. The generality of the setting raises some complications those authors handle with additional conditions on the measurement operator. For example, since the union of subspaces may not be...
closed under addition, the RICs must be extended to sums of vectors from the union of subspaces. In [24], Hegde et al. establish the need for an exact projection method onto the model and further analyze IHT and CoSaMP in this setting.

One obviously important example of a union of subspaces is the standard compressed sensing regime (1). Consider the $k$-sparse vectors where each subspace $\chi_i$ is determined by a fixed index set $Q_i \subset \{1, \ldots, n\}$ with $|Q_i| = k$. Then $\chi_i = \{x \in \mathbb{R}^n : \text{supp}(x) \subseteq Q_i\}$ is a $k$-dimensional subspace and the set of all $k$-sparse vectors is

$$\Sigma_{k,n} = \left\{ \sum_{i=1}^{q} \chi_i \mid q = \binom{n}{k} \right\}.$$  

(7)

A second example, relevant to the following discussion, is the well studied block sparsity. Suppose $x \in \mathbb{R}^n$ is partitioned into $r$ blocks $x_i$ each of length $n_i$, so that $x = [x_1|x_2| \cdots |x_r] \in \mathbb{R}^{\Sigma r}$ with $n = \sum_{i=1}^{r} n_i$. The vector $x$ is $s$-block sparse if at most $s$ of the blocks are nonzero and the remaining $r - s$ blocks are zero. Typically, in this setting, $n_i = n/r$ is constant. This model, too, can be expressed as a union of subspaces. Fix an $s$-block sparse vector $\tilde{x}$, whose support $\tilde{Q} \subset \{1, \ldots, n\}$ is no larger than $\tilde{k} = \sum_{i \in \Lambda} n_i$ where $\Lambda = \{j : \tilde{x}_j \neq 0\}$. When the blocks have equal sizes, we have $\tilde{k} = \frac{r}{s}n$, and when the sizes vary there is clearly a maximum $\tilde{k}$ for the model. In either case, the vector $\tilde{x}$ lies in the $\tilde{k}$-dimensional subspace $\chi_j = \{x \in \mathbb{R}^n : \text{supp}(x) \subseteq Q_j\}$ and the set of all $s$ block sparse vectors is the union of all possible such $\tilde{k}$-dimensional subspaces.

\section*{B. SPARSITY IN LEVELS}

Recently, much interest has been given to a few highly-related, specific classes of union of subspaces models. For example, iterative greedy algorithms IHT, CoSaMP, and Hard Threshold Pursuit (HTP) [18] have recently been studied in the following models. With some notational and organizational choices, we present these models as defined in the respective articles.

1) SPARSITY IN LEVELS [25], [26]

Let $z \in \mathbb{R}^n$. Then $z$ is $(k, M)$-sparse in levels if

- we define the sparsity levels $M = (M_1, \ldots, M_r)$ where $1 \leq M_1 < M_2 < \cdots < M_r = n$,
- we define the local sparsities $k = (k_1, \ldots, k_r)$ where $k_j \leq M_j - M_{j-1}$ with $M_0 = 0$, and
- $|\text{supp}(z) \cap \{M_{j-1} + 1, \ldots, M_j\}| \leq k_j$ for $i = 1, \ldots, r$.

The sparsity in levels model provides theoretical foundations for the use of practical compressed sensing processes such as using Fourier measurements of images sparse in the wavelet domain [25], [27]. By capturing the structure of these imaging processes, the sparsity in levels model permits theory that explains high quality sparse approximation performance even for imaging techniques that fail to satisfy assumptions from the standard sparsity model. There is now a robust set of theoretical advances along these lines [25], [26], [27], [28], [29].

2) HIERARCHICAL SPARSITY [30], [31], [32]

Let $z = (z_1, \ldots, z_r) \in \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_r}$. Let $\kappa = (\kappa_1, \ldots, \kappa_r)$, then $z$ is $(s, k)$-hierarchically sparse if

- at most $s$ blocks $z_i$ are nonzero, and
- each nonzero block $z_i$ is $\kappa_i$-sparse.

When $\kappa_i = k$ is fixed, the hierarchical sparsity model has also been studied as bivel feature selection in [33]; iterative thresholding algorithms for bivel feature selection were studied in [34], [35].

3) SPARSITY WITH STRUCTURED SUPPORT [36], [37]

Let $z \in \mathbb{R}^n$. Let $Q_1, \ldots, Q_r \subset \{1, \ldots, n\}$ be disjoint index sets. Then $z$ is $k$-sparse with structured support if

- there exist local sparsities $\kappa_1, \ldots, \kappa_r$ with $\kappa = \sum_{i=1}^{r} \kappa_i$,
- and $z$ has at most $\kappa_i$ nonzeros when restricted to the index set $Q_i$, i.e. $\|z|_{Q_i}\| \leq \kappa_i$.

While the different points of view can be helpful, these three sparsity models are essentially equivalent. Clearly the sparsity in levels model is a specific instance of a structured support. Also, the knowledge of the structured support allows us to consider a vector as being sparse in levels with the introduction of a known permutation matrix. Finally, there is a one-to-one mapping of hierarchically sparse vectors $z = (z_1, z_2, \ldots, z_r) \in \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_r}$ with $|z| \leq k$ to sparse in levels vectors $x = [x_1|x_2| \cdots |x_r] \in \mathbb{R}^n$ with $n = \sum_{i=1}^{r} n_i$. To require that only $s$ blocks $x_i$ of a sparse in levels vector $x$ be nonzero, we simply set $\kappa_j = 0$ for all but $s$ values of $j \in \{1, \ldots, r\}$.

A helpful and common property linking these models together is the ability to explicitly define the index sets for the support from the structured support formulation.

Definition 2 (Structured Support Model): We say a union of $k$-dimensional subspaces $\Sigma = \cup_{j \in \Lambda} \mathbb{R}^{\Sigma}$ is a structured support model if for any vector $x \in \Sigma$ there exists an index set $Q \subset \{1, \ldots, n\}$ such that $x = (x)_{Q}$ and $|Q| \leq k$.

One advantage of restricting our attention to structured support models is the simple way we can deal with sums of elements from such models. While the more general models of [21], [22], [23], [24] require careful tracking of direct sums of subspaces, direct sums are simply captured by a union of index sets in a structured support model.

Lemma 1: Let $\Sigma$ be a structured support model and let $z_1, \ldots, z_r \in \Sigma$ with $z_0 = (z_i)_{Q_i}$ and $|Q| \leq k$ for each $i$. Then $z_1 + z_2 + \cdots + z_r = (z_1 + z_2 + \cdots + z_r)_{Q}$ for $Q = \cup_{i} Q_i$ and $|Q| \leq ck$.

This observation allows us to generalize the definitions of restricted isometry properties and mimic the analysis of algorithms from the standard $k$ sparse model $\Sigma_{k,n}$ with essentially no major effort.

Now, in order to avoid introducing yet another name for these equivalent models, we adopt the sparsity in levels language to support further understanding the variety of sensing and sparsity combinations that perform well in practice.

Definition 3 (Sparsity in Levels Model [25], [26]): Let $x \in \mathbb{R}^n$. For each $i \in \{1, \ldots, r\}$ define the index sets
$S_i = \{n_{i-1} + 1, \ldots, n_i\} \subset \{1, \ldots, n\}$ where $0 = n_0 < n_1 < n_2 < \cdots < n_r = n$, and let $k_i \in \mathbb{N}$. Define the levels $n = (n_1, \ldots, n_r)$ and the local sparsities $k = (k_1, \ldots, k_r)$. Then $x$ is $(k, n)$-sparse if $||x||_0 \leq k_i$ for each $i \in \{1, \ldots, r\}$.

We refer to $k = \sum_{i=1}^r k_i$ as the (total) sparsity of $x$.

The ability to view each of these models as structured support models allows us to consider them as a union of subspaces. If we fix a set of local sparsities and sparsity levels $(k, n)$, let $k = \sum_{i=1}^r k_i$, and let $Q_i = \bigcup_{i=1}^r R_{i,j}$ where $R_{i,j} \subseteq S_i$ and $|R_{i,j}| \leq k_i$. Then $x_{\text{loc}} = \{x \in \mathbb{R}^n : \supp(x) \subset Q_i\}$ is a $k$-dimensional subspace. The set of all $(k, n)$-sparse vectors is

$$\Sigma_{(k,n)} = \bigcup_{i=1}^q x_j with q = \prod_{i=1}^r \binom{n_i}{k_i}. \tag{8}$$

For each vector $x \in \Sigma_{(k,n)}$, its support is denoted by a set $Q \subset \{1, \ldots, n\}$ with $k_i$ indices in each of the partitions $(n_{i-1}+1, \ldots, n_i)$; we denote the set of all such support sets $\Delta_{(k,n)}$.

These models benefit from the structure of the support sets within the union of subspaces. Suppose we have fixed the dimension $n$, levels $n$, and local sparsities $k$ for the union of subspaces, $\Sigma_{(k,n)}$. For any two support sets $Q, S \in \Delta_{(k,n)}$, the union $Q \cup S$ also has structure; in each partition $(n_{i-1} + 1, \ldots, n_i)$, $Q \cup S$ has at most $2k_i$ values. In fact, a union of $c$ such sets $T = Q_1 \cup Q_2 \cup \cdots \cup Q_c$ has at most $ck$ values in $S_i = (n_{i-1}+1, \ldots, n_i)$. In other words, $T \in \Delta_{(k,n)}$ where each component of the $r$-tuple $ck$ is the minimum of $ck_i$ and $n_i - n_{i-1}$. Likewise, if $x^1, x^2, \ldots, x^c \in \Sigma_{(k,n)}$ then a linear combination $x = \sum \omega_i x^i \in \Sigma_{(k,n)}$.

Notice that $(k, n)$-sparse vectors are $k$-sparse vectors, i.e. $\Sigma_{(k,n)} \subset \Sigma_{k,n}$ for $k = \sum_{i=1}^r k_i$. As a result, the theory from standard compressed sensing on sparse vectors and from model based compressed sensing apply directly to the sparsity in levels setting. By exploiting the structure of sparse in levels vectors, algorithms are able to recover vectors with a higher total sparsity using the same number of measurements [26], [32], [37], while the model also provides the theoretical foundation for recovery success in Fourier imaging and other coherent measurement processes [25], [27], [28].

### C. GREEDY ALGORITHMS FOR STRUCTURED SPARSITY

For $A \in \mathbb{R}^{m \times n}$ and $x \in \Sigma_{k,n}$, the measurements $y = Ax$ have at least an $(n - m)$-dimensional preimage $\{z \in \mathbb{R}^n : y = Az\}$. Therefore, the sparsity cost function (1) or constraint (2) help limit the search for the vector $x$. In fact, once the support of $x$ is known, the reconstruction problem is overdetermined and can be accomplished by projection. As such, the greedy approach to solving (1) and (2) is to find the supporting subspace of $x$.

For the standard $k$-sparse model, the best supporting subspace of a vector $w \in \mathbb{R}^n$ is the space indexed by the largest magnitudes of the entries in $w$. Similarly, to approximate $w$ with a vector with local sparsities $k$ and sparsity levels $n$, we need the indices of the largest magnitudes of $w$ on each level. We define $\text{DetectSupport}_l(w)$ to be the index set of $k$ largest entries of $|w|$. Furthermore, to find the best supporting set from $\Delta_{(k,n)}$, we let $\text{DetectSupport}_{(k,n)}(w)$ be the index set which, for each $i = 1, \ldots, r$, contains the indices of the $k_i$ largest magnitudes of $w$ on the level $S_i = \{n_{i-1} + 1, \ldots, n_i\}$. In other words, we have

$$\text{DetectSupport}_{(k,n)}(w) = \bigcup_{i=1}^r \text{DetectSupport}_{k_i}(w) \big|_{S_i}.$$ 

The basic outline of an iterative greedy algorithm is to conduct a line search update, identify the best support set after the update, and then restrict the update to this supporting space either via hard threshold or projection. For example, if the current $k$-sparse approximation is $x_{l-1}$, the most basic iterative hard thresholding uses the residual, $r_{l-1} = A^*(y - Ax_{l-1})$ as the search direction so that

$$T_l = \text{DetectSupport}_k(x_{l-1} + r_{l-1}) \tag{9}$$

$$x_l = (x_{l-1} + r_{l-1})_{T_l}. \tag{10}$$

A two-stage algorithm, such as CoSaMP, introduces an intermediate optimization problem over a (possibly larger) subspace:

$$Q_l = \text{DetectSupport}_{2k}(r_{l-1}) \tag{11}$$

$$w_l = \arg \min_{z \in \supp(x) \subset Q_l \cup T_{l-1}} \|y - Az\|_2 \tag{12}$$

$$T_l = \text{DetectSupport}_k(w_l) \tag{13}$$

$$x_l = (w_l)_{T_l}. \tag{14}$$

Both of these algorithms have been extended to the sparsity in levels setting [26], [37] by simply utilizing $\text{DetectSupport}_{(k,n)}(x_{l-1} + r_{l-1})$ in (9), $\text{DetectSupport}_{(k,n)}(r_{l-1})$ in (11), and $\text{DetectSupport}_{(k,n)}(w_l)$ in (13). In fact, in [26], the study includes empirical performance comparisons with normalized iterative thresholding [15] whereby the line search uses a step size selected in each iteration so that if $T_{l-1}$ is the correct support set, then NIHT will take the optimal steepest descent step on that space. The authors pose as a future problem the need to analyze NIHT in the sparsity in levels paradigm.

Given that NIHT was seen to be competitive on $k$-sparse vectors [15], [38] and was then observed to also perform admirably in the sparsity in levels setting [26], we provide the extension of the theoretical recovery guarantees for NIHT.

Moreover, as the large scale empirical testing in [38] lead to the development of CGIHT and CGIHT Restarted by exploiting the advantages of multiple iterative greedy algorithms, we also provide the extension of CGIHT Restarted to the sparsity in levels model. Similar to the theory in the standard compressed sensing model, the theoretical recovery guarantee has

---

1The similar Hard Thresholding Pursuit was studied in [32].
Algorithm 1: NIHT.

Initialization: Set $w^{-1} = A^*y$, $(k, n)$

$T^0 = \text{DetectSupport}_{(k,n)}(w^{-1})$,

$x^0 = (w^{-1})_r$, and $l = 1$.

iteration: During iteration $l$, do

1: $r^{l-1} = A^* (y - Ax^{l-1})$ (compute the residual)

2: $\alpha^{l-1} = \frac{\|r^{l-1}\|_2^2}{\|A r^{l-1}\|_2^2}$ (optimal stepsize on $T^{l-1}$)

3: $w^{l-1} = x^{l-1} + \alpha^{l-1} r^{l-1}$ (update along search direction)

4: $T^l = \text{DetectSupport}_{(k,n)}(w^{l-1})$ ($(k,n)$-structured support set of $k$ largest entries)

5: $x^l = (w^{-1})_{T^l}$ (restriction to support set $T^l$)

Algorithm 2: CGIHT Restarted.

Initialization: Set $T^{-1} = \emptyset$, $p^{-1} = 0$, $w^{-1} = A^*y$, $(k, n)$

$T^0 = \text{DetectSupport}_{(k,n)}(w^{-1})$,

$x^0 = (w^{-1})_r$, and $l = 1$.

iteration: During iteration $l$, do

1: $r^{l-1} = A^* (y - Ax^{l-1})$ (compute the residual)

2: if $T^{l-1} \neq T^{l-2}$ (set orthogonalization weight)

$\beta^{l-1} = 0$ (restart)

else

$\beta^{l-1} = \frac{\|r^{l-1}\|_2^2}{\|A r^{l-1}\|_2^2}$ (orthogonalization weight)

3: $p^{l-1} = r^{l-1} + \beta^{l-1} p^{l-2}$ (define the search direction)

4: $\alpha^{l-1} = \frac{\|r^{l-1}\|_2^2}{\|A p^{l-1}\|_2^2}$ (optimal stepsize on $T^{l-1}$)

5: $w^{l-1} = x^{l-1} + \alpha^{l-1} p^{l-1}$ (update along search direction)

6: $T^l = \text{DetectSupport}_{(k,n)}(w^{l-1})$ ($(k,n)$-structured support set of $k$ largest entries)

7: $x^l = (w^{-1})_{T^l}$ (restriction to support set $T^l$)

only been proven for CGIHT Restarted wherein the conjugate gradient structure is restored when the support set changes. However, the empirical performances of both CGIHT and CGIHT Restarted are competitive with CoSaMP in which vectors are recovered, but both have a significantly reduced computational complexity comparable to NIHT.

The algorithm CGIHT is the same as Algorithm 2, except the orthogonalization weight is never set to zero in Step 2.

III. RECOVERY GUARANTEES FOR SPARSITY IN LEVELS

As mentioned in Section I, most recovery guarantees in compressed sensing and sparse approximation come in the form of a restricted isometry property. The analysis and results typically look cleaner when a single, symmetric restricted isometry constant is used in the analysis, though these cleaner results often mask the important asymmetric scaling impacts of a matrix. In this paper, we will utilize the more general asymmetric restricted isometry constants.

Definition 4 (Asymmetric Restricted Isometry Constants [21], [29], [39]): Let $A$ be an $m \times n$ matrix and fix levels $n$ and local sparsities $k$. The asymmetric $(k, n)$-restricted isometry constants $L((k, n), A)$ and $U((k, n), A)$ for $(k, n)$-sparse vectors $x \in \Sigma_{(k,n)}$ are defined as:

\[
L_k = L((k, n), A) := \min_{c \geq 0} \frac{\|Ax\|_2^2}{\|x\|_2^2} \leq (1 - c) \quad \text{for all } x \in \Sigma_{(k,n)}; \\
U_k = U((k, n), A) := \min_{c \geq 0} \frac{\|Ax\|_2^2}{\|x\|_2^2} \geq (1 + c) \quad \text{for all } x \in \Sigma_{(k,n)}.
\]  

In [26], Adcock et al. pose as a future problem the need to analyze NIHT for the sparsity in levels problem. The first step in doing so is to bound the step sizes from each iteration. Fortunately, any matrix with appropriate RICs will admit a stable behavior on the step sizes.

Lemma 2 ([40]): Let $k$ and $n$ define local sparsities and sparsity levels and fix the sparse in levels model $\Sigma_{(k,n)}$. Suppose $A \in \mathbb{R}^{m \times n}$ has $(k, n)$-restricted isometry constants $L_k := L((k, n), A)$ and $U_k := U((k, n), A)$. Let $\alpha^l$ be the step size generated in iteration $l + 1$ of NIHT, Algorithm 1, Step 2. Then

\[
\frac{1}{1 + U_k} \leq \alpha^l \leq \frac{1}{1 - L_k}. 
\]

Proof: Fix the iteration $l + 1$ in Algorithm 1 and let $r^l = A^* (y - Ax^l)$ where $y$ is the input measurements and $x^l \in \Sigma_{(k,n)}$ is the approximation from the previous iteration. Let $T^l \in \Sigma_{(k,n)}$ be the support set of $x^l$. Then, restricting to $T^l$ we see that $(r^l_{T^l}) \in \Sigma_{(k,n)}$. From Definition 4, we have

\[
1 - L_k \leq \frac{\|A(r^l_{T^l})\|_2^2}{\|r^l_{T^l}\|_2^2} = \frac{1}{\alpha^l} \leq 1 + U_k,
\]

which is equivalent to (17).

The proof of Lemma 2 is nearly identical to that in [40] with the only change being the RICs on the structured support model $\Sigma_{(k,n)}$. In the appendix, we combine the proofs of [41, Thm. 6.18] and [40, Thm. 2] to obtain the guarantee for NIHT. This proof for the sparsity in levels model includes the standard sparsity result when considering only a single, $k$-sparse level.

Theorem 1 (NIHT [40]): Let $k$ and $n$ define local sparsities and sparsity levels and fix the sparse in levels model $\Sigma_{(k,n)}$. Suppose $A \in \mathbb{R}^{m \times n}$ has $(k, n)$-restricted isometry constants $L_k := L((k, n), A)$ and $U_k := U((k, n), A)$, $c = 1, 2, 3$. Let $y = Ax + e$ for some $x \in \Sigma_{(k,n)}$ and $e \in \mathbb{R}^m$ an additive misfit. Define

\[
\mu = \sqrt{3}\left(\frac{U_{3k} + L_{3k}}{1 - L_k}\right) \quad \text{and} \quad \xi = \sqrt{3}\left(\frac{1 + U_{2k}}{1 - L_k}\right). 
\]

If

\[
\sqrt{3}\left(\frac{U_{3k} + L_{3k}}{1 - L_k}\right) < 1. 
\]
then (20) ensures \( \mu < 1 \) and
\[
\|x^l - x\|_2 \leq \mu^l \|x^0 - x\|_2 + \frac{\xi}{1 - \mu} \|e\|_2
\]
(21)
where \( x^l \) is the \( l \)th iterate of NIHT.

CGIHT performs a conjugate gradient projection on a given support set. When the algorithm decides the support should change, CGIHT Restarted uses the residual as the search direction thereby restarting the conjugate gradient method. This assures the conjugate orthogonality of the search directions while on a fixed support. This orthogonality of the search directions also admits an analysis of the algorithm which calls upon several conjugate gradient properties. The first such situation relates the residual to the conjugate search direction and allows us to bound the step size and orthogonalization weights, as we did with NIHT.

**Lemma 3 ([19]):** Let \( k \) and \( n \) define local sparsities and sparsity levels and fix the sparse in levels model \( \Sigma_{(k, n)} \). Suppose \( A \in \mathbb{R}^{m \times n} \) has \( (k, n) \)-restricted isometry constants \( L_k := L((k, n), A) \) and \( U_k := U((k, n), A) \). Let \( \alpha' \) be the step size and \( \beta' \) be the orthogonalization weight generated in Step 4 and 2, respectively, of iteration \( l + 1 \) of CGIHT Restarted, Algorithm 2. Then
\[
\frac{1}{1 + U_k} \leq \alpha' \leq \frac{1}{1 - L_k},
\]
(22)
\[
|\beta'| \leq \frac{(L_k + U_k)(1 + U_k)}{(1 - L_k)^2}.
\]
(23)

**Proof:** Similar to the proof of bounding the step sizes in NIHT, this proof directly follows the existing argument in [19, Lemma 4.2], but with the support sets and RICs having the sparsity in levels structure. When the support changes from iteration to the next, i.e. \( T^l \neq T^{l-1} \), the step size \( \alpha' \) is the same as in NIHT and is bounded directly as in the proof of Lemma 2.

When the support set is unchanged from one iteration to the next, \( T^l = T^{l-1} \), CGIHT Restarted uses step sizes and search directions identical to those in a conjugate gradient method restricted to \( T^l \). Therefore, the various properties of the conjugate gradient search directions yield two important inequalities:
\[
\|A(p')_T^l\|_2 \leq \|A(r')_T^l\|_2,
\]
(24)
\[
\|r'_T^l\|_2 \leq \|p'_T^l\|_2.
\]
(25)
These two inequalities provide a straightforward path to (22), as
\[
\frac{1}{\alpha'} \leq \frac{\|A(p')_T^l\|_2}{\|r'_T^l\|_2} \leq \frac{\|A(r')_T^l\|_2}{\|r'_T^l\|_2} \leq (1 + U_k),
\]
and
\[
\alpha' \leq \frac{\|r'_T^l\|_2^2}{\|A(p')_T^l\|_2^2} \leq \frac{\|p'_T^l\|_2^2}{\|A(p')_T^l\|_2^2} \leq \frac{1}{1 - L_k}.
\]

Now, the orthogonalization weight \( \beta' \) requires a little more finesse and the use of Lemma 5 stated and proved in the appendix. To begin with, we note that the definition of the orthogonalization weight \( \beta' \) has an equivalent form [42, p. 35], namely
\[
\beta' = -\frac{\langle r'_T^l, (A^*A(p'^l)_T^l)_{T^l} \rangle}{\|A(p'^l)_T^l\|_2^2}.
\]
(26)
Since \( r' = A^*(y - Ax') \) and \( x' = (x^l + \alpha'^l p'^l)_{T^l} \), we have
\[
(r'_T^l)_{T^l} = (A^*(y - Ax'^l))_{T^l} - \left(\alpha'^l A^*A(p'^l)_{T^l} \right)_{T^l} = (r^l)_{T^l} - \left(\alpha'^l A^*A(p'^l)_{T^l} \right)_{T^l}.
\]
(27)
From the definition of the search direction, we see that \( r^{l-1} = p^{l-1} - \beta^{l-1}p^{l-2} \). Using this description of \( r^{l-1} \) we can use (27) to express the current residual \( (r'_T^l)_{T^l} \) in terms of the two prior search directions, namely
\[
(r'_T^l)_{T^l} = \left( (I - \alpha^{l-1}A^*A)p^{l-1} \right)_{T^l} - \beta^{l-1}(p^{l-2})_{T^l}.
\]
(28)
The entire point of the conjugate gradient search is that the search directions are conjugate orthogonal so that
\[
\left( \beta^{l-1}(p^{l-2})_{T^l}, (A^*A(p^{l-1})_{T^l})_{T^l} \right) = 0.
\]
(29)
Therefore, we can use (28) and (29) to update the numerator of (26) as
\[
\beta' \leq -\frac{\left( \langle I - \alpha^{l-1}A^*A(p^{l-1})_{T^l}, (A^*A(p^{l-1})_{T^l})_{T^l} \rangle \right)}{\|A(p^{l-1})_{T^l}\|_2^2}.
\]
(30)
Finally, applying the Cauchy-Schwarz inequality and Lemma 5 to the numerator in (30) while invoking the definition of the lower asymmetric RICs in the denominator, we have
\[
|\beta'| \leq \frac{L_k + U_k}{1 - L_k} \left( \frac{(1 + U_k)^2(1 - U_k)}{(1 - L_k)^2} \right)^{1/2}.
\]
(31)

Having bounded the step sizes and orthogonalization weights using restricted isometry constants, we can provide a sufficient restricted isometry property for guaranteed recovery of sparse in levels vectors using CGIHT Restarted.

**Theorem 2 (CGIHT [19]):** Let \( k \) and \( n \) define local sparsities and sparsity levels and fix the sparse in levels model \( \Sigma_{(k, n)} \). Suppose \( A \in \mathbb{R}^{m \times n} \) has \( (ck, n) \)-restricted isometry constants \( L_{ck} = L((ck, n), A) \) and \( U_{ck} = U((ck, n), A) \), \( c = 1, 2, 3 \). Let \( y = Ax + e \) for some \( x \in \Sigma_{(k, n)} \) and \( e \in \mathbb{R}^m \) an additive misfit. Define
\[
\mu = \frac{1}{2} \left( \tau_1 + \sqrt{\tau_1^2 + 4\tau_2} \right) \quad \text{and} \quad \xi = \sqrt{3} \left( \frac{\sqrt{1 + U_{ck}}}{1 - L_{ck}} \right),
\]
(31)
where
\[
\tau_1 = \sqrt{3} \frac{U_{3k} + L_{3k}}{1 - L_k} + \frac{(1 + U_k)(U_k + L_k)}{(1 - L_k)^2},
\]
\[
\tau_2 = \sqrt{3} \frac{(1 - L_{3k})(1 + U_k)(U_k + L_k)}{(1 - L_k)^3}.
\]
If
\[
\left( \frac{U_{3k} + L_{3k}}{(1 - L_k)^2} \right) \left( (1 + 2\sqrt{3}) - \sqrt{3}L_k + (1 + \sqrt{3})U_k \right) < 1,
\]
then (34) ensures \( \mu < 1 \) and
\[
\|x^\ell - x\|_2 \leq \mu^\ell \|x^0 - x\|_2 + \frac{\xi}{1 - \mu} \|e\|_2
\]
have \( k = 512 \). Empirical testing with various values of \( n \) and \( k \) yield results painting the same overall picture.

**B. DEGREE OF SPARSITY IN LEVELS STRUCTURE**

For Fig. 1, the vectors are all given four uniform levels \( \mathbf{n}_1 = (512, 1024, 1536, 2048) \). The experiments are conducted for two sparsity patterns with proportions \((3/8, 1/8, 3/8, 1/8)\) and \((1/2, 0, 1/2, 0)\) yielding local sparsities \( \mathbf{k}_1 = (192, 64, 192, 64) \) and \( \mathbf{k}_2 = (256, 0, 256, 0) \).

In Fig. 1(a) the vector has a moderate structure \( (\mathbf{k}_1, \mathbf{n}_1) \) and the algorithms are given the exact structure information with \( \tilde{\mathbf{k}} = \mathbf{k}_1 \), \( \tilde{\mathbf{n}} = \mathbf{n}_1 \). Even with only a moderate structure in the vectors, the algorithms recover the vectors with fewer measurements when given the exact information. In Fig. 1(b) and (c), the input vectors are created with highly structured local sparsities \( \mathbf{k}_2 \). In (b), the levels aware algorithms are given partial information: a two level structure \( \tilde{\mathbf{n}} = (512, 2048) \) with \( \tilde{\mathbf{k}} = (256, 256) \). Even with missing information, the levels aware algorithms still outperform the standard algorithms, although modestly. In (c), when given the exact sparsity, \( \tilde{\mathbf{k}} = \mathbf{k}_2 \) and \( \tilde{\mathbf{n}} = \mathbf{n}_1 \), the levels aware algorithms significantly outperform the standard algorithms. These three experiments indicate that both the significance of the structure in the vectors and accuracy of the information provided to the algorithms impact the performance of the sparsity in levels versions of the algorithms. While this matches the observations in [26], here we also see that CGIHTL and CGIHTL Restarted are competitive with CSMPPL and clearly outperform NIHTL.

**C. QUALITY OF ALGORITHM KNOWLEDGE**

In the following experiments, we demonstrate the impact of the quality of information provided to the algorithms. First of all, we observe that when a sparsity in levels adapted algorithm is provided incorrect information about the structure of the vector, the algorithm fails to recover the sparse in levels vector. The experiment depicted in Fig. 2 has vectors from \( \Sigma_{(\mathbf{k}_1, \mathbf{n}_1)} \) where \( \mathbf{n}_1 = (512, 1024, 1536, 2048) \) has four equal sized levels and the local sparsities are \( \mathbf{k}_1 = (192, 64, 192, 64) \) as before. However, in this experiment, the algorithm is given the incorrect sparsity in levels structure where \( \tilde{\mathbf{n}} = (512, 2048) \) yet \( \tilde{\mathbf{k}} = (256, 256) \). Since the vectors drawn from \( \Sigma_{(\mathbf{k}_1, \mathbf{n}_1)} \) cannot have 256 nonzeros in the first 512 entries, the levels aware algorithms CGIHTL, CGIHTL Restarted, NIHTL, and CSMPPL all fail to recover the vectors. Note that the standard algorithms are unencumbered by the incorrect information and recover the \((\mathbf{k}_1, \mathbf{n}_1)\)-sparse in levels vectors.

We now run an experiment with a more substantial sparsity in levels structure. The sparse in levels vectors for the experiments in Fig. 3 all have eight uniform levels \( \mathbf{n}_3 = (256, 512, 768, 1024, 1280, 1536, 1792, 2048) \) with highly structured local sparsities \( \mathbf{k}_3 = (128, 0, 128, 0, 128, 0, 128, 0) \). The first experiment, whose results are represented in Fig. 3(a), provides essentially no valuable information to the levels adapted algorithms. In this case, the algorithms are given an accurate but uninformative four level structure with \( \tilde{\mathbf{n}} = (512, 1024, 1536, 2048) \) and local
FIGURE 3. Compressed sensing (1) for \( n = 2048\) and \( k = 512\). Three experiments on quality of information given to the sparsity in levels algorithms. All experiments use 8 uniform levels \( n_3 = (256, 512, 768, 1024, 1280, 1536, 1792, 2048)\) with local sparsities \( k_3 = (128, 0, 128, 0, 128, 0, 128, 0)\). (a) Minimal information is provided to the algorithms with \( \tilde{n} = (512, 1024, 1536, 2048)\) and \( \tilde{k} = (128, 128, 128, 128, 128, 128)\); the levels algorithms exhibit no advantage. (b) Partial information is provided to the algorithms with \( \tilde{n} = (256, 1024, 1280, 2048)\) and \( \tilde{k} = (128, 128, 128, 128, 128, 128)\); the levels algorithms exhibit some advantage. (c) Exact sparsity in levels information \( (k_3, n_3)\) is provided to the levels algorithms; the levels algorithms show substantial advantage over the standard algorithms. In all cases, CGIHT and CGIHT Restarted are competitive with CSMPSP and outperform NIHT.

FIGURE 4. Compressed sensing (1) for \( n = 2048\), \( k = 512\). Dyadic Levels \( n_4 = (128, 256, 512, 1024, 2048)\) with sparsity in levels algorithms receiving exact sparsity in levels structure: (a) \( k_4 = (128, 64, 128, 128, 64)\); (b) \( k_5 = (128, 128, 112, 96, 48)\).

sparsities \( \tilde{k} = (128, 128, 128, 128)\). This lack of quality information for the algorithms results in no performance improvement; the structure aware algorithms perform exactly like the standard algorithms.

In the next experiment, we still provide the algorithms with only a four level structure, but now the structure more closely matches the actual structure of the input vectors. In Fig. 3(b), the algorithms are provided with the levels \( \tilde{n} = (256, 1024, 1280, 2048)\) and local sparsities \( \tilde{k} = (128, 128, 128, 128)\). With this limited information, the levels aware algorithms use the knowledge to modestly outperform their standard counterparts.

Finally, when the algorithms are provided the exact eight level sparsity in levels structure \( \tilde{n} = n_3\) and \( \tilde{k} = k_3\), we see in Fig. 3(c) that the algorithms are able to exploit this knowledge to considerably outperform the naive standard versions of the algorithm.

These four experiments show that the sparsity in levels adapted algorithms are highly dependent on the quality of the structure information provided to the algorithm. Moreover, we see that for all experiments, the CGIHT family of algorithms has performance comparable to CSMPSP.

D. DYADIC STRUCTURE

One important idea behind structured sparsity [23] and the motivation for work on the sparsity in levels model [25], [27] is the use of algorithms for sparse recovery of wavelet coefficients. In Fig. 4, the vectors are created with a dyadic levels structure \( n_4 = (128, 256, 512, 1024, 2048)\) where the five levels have proportions \((1/16, 1/16, 1/8, 1/4, 1/2)\) of the total length. We conduct two experiments where the algorithms are provided the exact sparsity in levels structure. In Fig. 4(a) the vector has limited decay in coefficients per level with \( k_4 = (128, 64, 128, 128, 64)\). On the other hand, a more significant coefficient decay modeling a wavelet compression of a signal is shown in (b) where \( k_5 = (128, 128, 112, 96, 48)\).

In both experiments, the levels aware algorithms are given the exact sparsity in levels information. Again, we see here that the standard algorithms have a recovery phase transition at roughly the same number of measurements \( m\). However, the
levels aware algorithms gain a significant advantage with the exact information and perform much better in (b) on the more highly structured sparsity in levels pattern; this observation matches the theoretical discussions and asymptotic sparsity in [25], [27]. Here again, the CGIHT family of algorithms significantly outperforms NIHT and matches the performance of CSMPSP.

E. PHASE TRANSITIONS AND ALGORITHM SELECTION

So far our empirical investigation has made a case that the standard and levels aware versions of CGIHT and CGIHT Restarted are comparable in performance to CSMPSP. In this section we will more clearly indicate which algorithm should be used for a given problem. First, we recall the notion of an empirical phase transition for sparse approximation algorithms. Given a problem class with an $m \times n$ matrix $A$ and a vector $x \in \mathbb{R}^n$ with $\|x\|_0 = k$, we define the undersampling and oversampling ratios

$$\delta = \frac{m}{n}, \quad \rho = \frac{k}{m}. \quad (36)$$

Then since $k \leq m \leq n$, the unit square provides a $(\delta, \rho)$ phase space. A phase transition curve for an algorithm is a function $\rho(\delta)$ that defines a curve where the algorithm successfully solves problems with sampling ratios below the curve, but fails to solve problems with sampling ratios above the curve. An empirical 50% phase transition curve for an algorithm is a function $\rho^*(\delta)$ where an algorithm is observed to solve roughly half of all problems with sampling ratios along the curve. Below the curve, the algorithm succeeds with high probability and fails with high probability above the curve. The region around the curve where the algorithm transitions from always succeeding to always failing has a width proportional to $\sqrt{m}$.

To investigate a problem where the sparsity in levels aware variants of the algorithms have the greatest advantage, our final two experiments return to the highly structured sparsity in levels experiment from Fig. 3(c), namely $n_3 = (256, 512, 768, 1024, 1280, 1536, 1792, 2048)$ with local sparsities $k_3 = (128, 0, 128, 0, 128, 0, 128, 0)$. For Fig. 5 we present results where the sparsity in levels variants of the algorithms are given the exact information $\tilde{n}_3 = n_3, \tilde{k}_3 = k_3$. Fig. 5(a) presents the 50% phase transition curves for CGIHT, CGIHT Restarted, CSMPSP, and NIHT along with their sparsity in levels variants.
FIGURE 7. Ratios of algorithm recovery time to best recovery time for \( n = 2048 \) using 8 uniform levels \( n_3 = (256, 512, 768, 1024, 1280, 1536, 1792, 2048) \) with local sparsities \( k_3 = (128, 0, 128, 0, 128, 0, 128, 0) \). Exact sparsity in levels information \((k_3, n_3)\) is provided to the levels algorithms. Color scale maximum of 3 \( \times \) the fastest algorithm includes all ratios greater than 3. (a) CGIHT Restarted; (b) CGIHTL Restarted; (c) CGIHT; (d) CGIHTL; (e) CSMPSP; (f) CSMPSPL.
those of the other algorithms. When given the sparsity in levels information and the ability to utilize that knowledge, the phase transitions increase significantly. Here again we observe that CGIHTL and CGIHTL Restarted are competitive with CSMPSPL; all three algorithms have nearly identical phase transition curves when $\delta \leq 0.25$.

When attempting to solve a problem, it is usually desirable to employ the algorithm that will find an accurate solution with the least computational cost. In Fig. 5(b) we provide an algorithm selection map [38] which indicates which algorithm recovers the measured vector in the least amount of time. More specifically, for each ordered pair of sampling ratios $(\delta, \rho)$, the algorithm with the smallest average recovery time is indicated on the map. There are two important observations in Fig. 5(b). First, for the overwhelming majority of the phase space, an algorithm from the CGIHT family is the algorithm of choice with CSMPSP being selected either when it is the only algorithm that is likely to solve the problem (e.g. $(\delta, \rho) = (0.5, 0.5)$) or when the problem will only require one iteration (e.g. the smallest values of $\rho = 0.02$). Second, when the standard algorithm will succeed, i.e. when the sampling ratios fall below the phase transition curves of CGIHT Restarted, the standard algorithm will solve the problem in less time and should be selected.

The best run time (ms) among all algorithms is given in Fig. 6. The transition from using the standard algorithm to using a sparsity in levels aware algorithm is very clearly delineated. In Fig. 7, we see how each of CGIHT(L) Restarted, CGIHT(L), and CSMPSP(L) compare to the best run time. These plots show the ratio of the average recovery time for each algorithm to the best average recovery time. In order to compare plots while still distinguishing between small ratios near 1, the colorbar is fixed to a minimum ratio of 1 and a maximum of 3; all ratios greater than 3 appear as the bright yellow top of the colorbar. Based on the combination of the algorithm selection map (Fig. 5) and the ratio of recovery times to the best time (Fig. 7), CGIHT Restarted and CGIHTL Restarted appear to be the algorithms of choice.

V. CONCLUSION

When a priori knowledge of additional structure is available for solving a compressed sensing or sparse approximation problem, adapting greedy algorithms to utilize this information can be advantageous. In Section III, we provided a theoretical recovery guarantee for the sparsity in levels model for both NIHT and CGIHT Restarted. Since the standard compressed sensing model is simply a trivial instance of the sparsity in levels models, Theorems 1 and 2 apply to all sparsity in levels models and slightly improve upon existing restricted isometry properties for guaranteed recovery for both algorithms. From the empirical investigation in Section IV, we see that the average case performance of CGIHTL and CGIHTL Restarted are highly competitive. With the worst case recovery guarantees and the observed average case performance, we make the following recommendations when solving a compressed sensing problem with structured sparsity.

Given the backing of Theorem 2, the empirical observations similar to those in Figs. 5–7, and its stability to noise [45], we recommend the use of CGIHT Restarted and CGIHT for Sparsity in Levels. In the problem regime where CGIHT Restarted is likely to succeed, use CGIHT Restarted for compressed sensing regardless of the known structure information. When given a problem whose sampling ratios are close to or above the 50% phase transition curve for CGIHT Restarted, use the sparsity in levels variant of CGIHT Restarted. For other problems in the smaller region of the phase space where Fig. 5(b) indicates using CSMPSPL, use a sparsity in levels version of CoSaMP.

APPENDIX A

PROOFS

We prove Theorems 1–2 which extend similar results to the sparsity in levels setting. These proofs combine the existing proofs of [19], [40], [41] to obtain recovery guarantees for algorithms designed to utilize the structured support sets. Furthermore, since the sparsity in levels model includes the standard compressed model $\Sigma_s$, the results in this section are also improvements over those existing in the literature in that the relevant constants $\mu$ and $\xi$ have been reduced.

While the proofs in this appendix appear on the surface to focus on an analysis for the compressed sensing problem (1), the proofs extend naturally to the sparse approximation setting (2). Suppose $x \in \mathbb{R}^n$ and $T \in \Delta_{(k,n)}$ is the support of the largest entries in $x$ with sparsity levels $n$ and local sparsities $k$. Now assume $y = Ax + e$ for some misfit vector $e$. Simply define $\tilde{x} = (x)^T$ and $\tilde{e} = A(x)^T + e$. Then we have $y = Ax + \tilde{e}$ where $\tilde{x} \in \Sigma_{(k,n)}$ is sparse in levels and $\tilde{e}$ is an additive misfit vector. Further analysis will provide bounds on errors in other norms similar to those in [26] and [41, Section 6.3].

A. TECHNICAL LEMMAS

This first lemma is based on the structure of the algorithms, relies on the format for finding the support set of the update, and is not concerned with the matrix $A$ nor RICs. This version is based on the beginning of [41, Thm. 6.18].

Lemma 4: Let $x^{i+1}, x^i \in \Sigma_{(k,n)}$ have support sets $T^{i+1}, T^i$, respectively. Let $\alpha^i \in \mathbb{R}$ and $d^i \in \mathbb{R}^n$ be a step size and direction vector from one of the iterative algorithms. Then,

$$\|x^{i+1} - x\|_2 \leq \sqrt{3}\|(x^i - x + \alpha^i d^i)_{T^i\cup T^{i+1}}\|_2. \tag{37}$$

Proof: In each of the algorithms, with $w^i = x^i + \alpha^i d^i$ the index set $T^{i+1}$ has the form

$$T^{i+1} = \text{DetectSupport}_{(k,n)}(w^i)$$

and contains the indices for the $k_i$ largest magnitude entries of $w^i$ on level $n_{i-1} + 1$ to $n_i$. Since both $T^{i+1}, T \in \Delta_{(k,n)}$, we

\footnote{The differences in recovery times between CGIHT(L) and CGIHT(L) Restarted are minor as seen in Fig. 7.}
have \( \|w^j\|_2^2 \leq \|w^j\|_{T_{j+1}}^2 \) and eliminating shared values gives
\[
\|w^j\|_{T_{j+1}}^2 \leq \|w^j\|_{T_{j+1}^1}^2.
\]
(38)

Now since \( x \) is zero on the index set \( T_{j+1} \),
\[
\|w^j\|_{T_{j+1}^1}^2 = \|w^j - x\|_{T_{j+1}^1}^2.
\]
(39)

On the other hand, \( x^{j+1} \) has no contribution on the index set \( T_{j+1} \), so that
\[
\|w^j\|_{T_{j+1}}^2 = \|w^j - x + x^{j+1}\|_{T_{j+1}}^2
\geq \|w^j - x\|_{T_{j+1}}^2
\]
\[
- \|w^j - x\|_{T_{j+1}^1}^2.
\]
(40)

Applying (39) to the right-hand side of (38), and (40) to the left,
\[
\|x - x^{j+1}\|_{T_{j+1}}^2 \leq \|w^j - x\|_{T_{j+1}^1}^2
\]
\[
+ \|w^j - x\|_{T_{j+1}^1}^2
\]
\[
\leq \sqrt{2} \|w^j - x\|_{T \setminus T_{j+1}}^2,
\]
(41)

where \( T \setminus T_{j+1} = (T \setminus T_{j+1}) \cup (j+1,T) \) is the symmetric difference of the sets.

Now notice that \( x^{j+1} = (w^j)_{T_{j+1}} \) so that
\[
x^{j+1} - x = (x^{j+1} - x)_{T_{j+1}} + (x^{j+1} - x)_{T \setminus T_{j+1}}
\]
\[
= (w^j - x)_{T_{j+1}} + (x^{j+1} - x)_{T \setminus T_{j+1}}.
\]
(42)

Thus, applying (41) to (42),
\[
\|x^{j+1} - x\|^2 \leq \|w^j - x\|_{T_{j+1}^1}^2 + 3\|x^{j+1} - x\|_{T_{j+1}^1}^2
\]
\[
\leq \|w^j - x\|_{T_{j+1}^1}^2 + 2\|w^j - x\|_{T \setminus T_{j+1}}^2
\]
\[
\leq 3\|x^{j+1} - x\|_{T \setminus T_{j+1}}^2.
\]
(43)

As usual with proofs involving RICs, we will use several bounds on matrices derived from the measurement matrix \( A \). These asymmetric restricted isometry constant (aRIC) bounds appear in various forms throughout the literature involving restricted isometry properties.

**Lemma 5:** Let \( S \subseteq \Delta_{k,n} \) and suppose \( A \in \mathbb{R}^{m \times n} \) has asymmetric RICs \( L_{ck}, \ U_{ck} \). Suppose \( \mathbf{z} \in \mathbb{R}^m \), \( \mathbf{e} \in \mathbb{R}^m \), and \( \alpha \in \mathbb{R} \) satisfies \( \frac{1}{1 + L_k} \leq |\alpha| \leq \frac{1}{1 - L_k} \). Then
\[
\|\alpha A_S^* \mathbf{e}\|_2 \leq \frac{1 + U_{ck}}{1 - L_k} \|\mathbf{e}\|_2;
\]
(44)
\[
\|I - \alpha A_S^* A_S\|_2 \leq \frac{L_{ck} + U_{ck}}{1 - L_k};
\]
(45)
\[
\|\left((I - \alpha A^* A)(\mathbf{z})\right)\|_2 \leq \frac{L_{ck} + U_{ck}}{1 - L_k} \|(\mathbf{z})\|_2;
\]
(46)
\[
\|\alpha A^* \mathbf{e}\|_2 \leq \frac{\sqrt{1 + U_{ck}}}{1 - L_k} \|\mathbf{e}\|_2.
\]
(47)

**Proof:** As defined in [39], the aRICs of \( A \) are determined by the extreme eigenvalues of the all Gram matrices from \( ck \)

B. NIHT

We put the technical lemmas together to prove Theorem 1 for NIHT.

**Proof of Thm. 1 (NIHT):** We assume that \( x \in \mathbb{R}^n \) has a structured support \( T \subseteq \Delta_{k,n} \). The algorithm is given \( y \) and \( A \) where \( y = Ax + e \).

Let \( r^j = A^*(y - Ax^j) \) be the residual in iteration \( j + 1 \) of NIHT, Algorithm 1, Step 1, and observe that the update in

\[
1 - L_{ck} = \min_{S} \lambda(A_S^* A_S),
\]
\[
1 + U_{ck} = \max_{S} \lambda(A_S^* A_S).
\]

In other words, \( 1 - L_{ck} \leq \lambda(A_S^* A_S) \leq 1 + U_{ck} \) for every eigenvalue of \( A_S^* A_S \). Thus (44) follows from applying the bound on \( \alpha \) and the standard operator norm inequality.

The eigenvalues of \( 1 - \alpha A_S^* A_S \) are bounded by
\[
1 - \alpha(1 + U_{ck}) \leq \lambda(I - \alpha A_S^* A_S) \leq 1 - \alpha(1 - L_{ck}).
\]

Applying the bound on the scalar \( \alpha \) and rearranging a bit, we have
\[
\frac{-(L_k + U_{ck})}{1 - L_k} \leq \lambda(I - \alpha A_S^* A_S) \leq \frac{U_k + L_{ck}}{1 + U_k}.
\]
(48)

Utilizing some basic inequalities on the aRIC, \( L_k \leq L_{ck}, U_k \leq U_{ck}, 1 - L_k \leq 1 - U_k \), we see that
\[
\frac{-(L_k + U_{ck})}{1 - L_k} \leq \frac{L_{ck} + U_{ck}}{1 - L_k}; \quad \text{and}
\]
\[
\frac{U_k + L_{ck}}{1 + U_k} \leq \frac{L_{ck} + U_{ck}}{1 - L_k}.
\]

Thus, we obtain (45) since
\[
\|I - \alpha A_S^* A_S\|_2 \leq \max_{\lambda} \lambda(I - \alpha A_S^* A_S) \leq \frac{L_{ck} + U_{ck}}{1 - L_k},
\]
(49)

and (46) follows directly from (45) by recognizing that restricting the matrix \( A \) to the columns indexed by \( S \) is equivalent to restricting the result of the matrix-vector product to the index set \( S \).

To obtain (47), we recognize that \( A_S^* A_S \) shares the same eigenvalues as \( A_S^* A_S \), so that \( \|A_S^* A_S\|_2 \leq 1 + U_{ck} \). Therefore
\[
\|\alpha A_S^* \mathbf{e}\|_2^2 = \langle \alpha A_S^* \mathbf{e}, \alpha A_S^* \mathbf{e}\rangle
\]
\[
= |\alpha|^2 \|\mathbf{e}\|_2 \|A_S^* A_S^* \mathbf{e}\|_2
\]
\[
\leq \left( \frac{1}{1 - L_k} \right)^2 \max_{\lambda} \lambda(A_S^* A_S^*) \|\mathbf{e}\|_2^2
\]
\[
\leq \frac{1 + U_{ck}}{(1 - L_k)^2} \|\mathbf{e}\|_2^2.
\]
(50)

\square
Step 3, uses $r^j$ as the search direction so that

$$x^j + \alpha^j r^j = x^j + \alpha^j A^*(y - Ax^j)$$
$$= x^j + \alpha^j A^*(x - x^j) + \alpha^j A^* e. \quad (51)$$

NIHT finds $T^{j+1} \in \Delta (k, n)$, and tracking the support restrictions in this iteration, we have

$$\left( (I - \alpha^j A^*) (x^j - x) + \alpha^j A^* e \right)_{T \cup T^{j+1}}$$
$$= \left( (I - \alpha^j A^*) (x^j - x) + \alpha^j A^* e \right)_{T \cup T^{j+1}}$$
$$= \left( (I - \alpha^j A^*) (x^j - x) + \alpha^j A^* e \right)_{T \cup T^{j+1}}$$

since $T \cup T^j$ contains the support of $x^j - x$.

Now, let $S = T \cup T^j \cup T^{j+1} \in \Delta (3k, n)$ and note $T \cup T^{j+1} \in \Delta (2k, n)$. Combining Lemma 4 with (52), applying the triangle inequality, and observing $(T \cup T^j), (T \cup T^{j+1}) \subset S$, we obtain

$$\| x^{j+1} - x \| \leq \sqrt{3} \| (I - \alpha^j A^*) (x^j - x) + \alpha^j A^* e \|_{T \cup T^{j+1}}$$
$$| \alpha^j A^* e + \| (I - \alpha^j A^*) (x^j - x) \|_{T \cup T^{j+1}} \|$$

$$\leq \sqrt{3} \| (I - \alpha^j A^*) (x^j - x) \|_{T \cup T^{j+1}} + \sqrt{3} \| \alpha^j A^* e \|_{T \cup T^{j+1}}$$

$$\leq \sqrt{3} \| (I - \alpha^j A^*) (x^j - x) \|_{T \cup T^{j+1}} + \sqrt{3} \| \alpha^j A^* e \|_{T \cup T^{j+1}} \quad (53)$$

Therefore, we can apply (46) and (47) from Lemma 5 to (53) to bound the error of a single iteration,

$$\| x^{j+1} - x \| \leq \mu \| x^j - x \| + \xi \| e \|, \quad (54)$$

where

$$\mu = \sqrt{3} \left( \frac{U_{3k} + L_{3k}}{1 - L_k} \right) \quad \xi = \sqrt{3} \left( \frac{\sqrt{1 + U_{2k}}}{1 - L_k} \right). \quad (55)$$

When $\mu < 1$, we arrive at the desired result,

$$\| x^{j+1} - x \| \leq \mu \| x^j - x \| + \xi \| e \|, \quad (56)$$

through an induction argument and by bounding a finite geometric sum of positive terms by its limit. □

C. CGIHT

The proof of Theorem 2 closely follows the proof from [19]. By using Lemma 4, the constants in the proof are improved. Of course the proof also assumes the sparsity in levels model. Lemma 6 provides the outline for the strategy where we attempt to establish a three-term recurrence relation on the errors from each iteration. By doing so, Lemma 6 ensures the approximations converge. The proof of this lemma is omitted.

Lemma 6 [19, Lemma 4.1]: Suppose $c_0, \eta, \tau_1, \tau_2 \geq 0$ and let $\mu = \frac{1}{2} (\tau_1 + \sqrt{\tau_1^2 + 4 \tau_2})$. Assume $c_1 \leq \mu c_0 + \eta$ and define $c_l = \tau_l c_{l-1} + \tau_2 c_{l-2} + \eta$ for $l \geq 2$. If $\tau_1 + \tau_2 < 1$, then $\mu < 1$ and

$$c_l \leq \mu^l c_0 + \eta \sum_{j=0}^{l-1} \mu^j. \quad (57)$$

Next, in the spirit of the conjugate gradient method, we isolate a portion of the proof bounding the error from one iteration in terms of the previous iterations.

Lemma 7 [19]: For any $j \in \mathbb{N}$, let $x^j \in \Sigma (k, n)$ have support set $T^j$, and let $\beta^j \in \mathbb{R}$ be an orthogonalization weight from CGIHT. Now suppose $x \in \Sigma (k, n)$ with support set $T$, and let $\alpha^{j+1} \in \mathbb{R}$ be a step size from CGIHT. Suppose $A$ is an $m \times n$ matrix with $m < n$ and let $y = Ax + e$ for some additive misfit $e \in \mathbb{R}^m$. If $w^{j-1}$ is the update vector from Algorithm 2, Step 5, then

$$\| (w^{j-1} - x)_{T \cup T^j} \| \leq \| (I - \alpha^{j+1} A^*) (x^j - x)_{T \cup T^j} \| + \| (\alpha^{j+1} A^* e)_{T \cup T^j} \| \quad (58)$$

Proof: First, iteratively applying the definition of the search directions allows us to write them entirely in terms of the residuals, namely

$$p^{j-1} = r^{j-1} + \sum_{i=0}^{l-2} \left( \prod_{i=j+1}^{l-1} \beta^i \right) r^i. \quad (59)$$

Considering the residuals in terms of the iterates, we have

$$r^{j-1} = A^* (x - x^{j-1}) + A^* e. \quad (60)$$

Therefore, we have

$$w^{j-1} - x = x^{j-1} - x + \alpha^{j-1} r^{j-1} + \alpha^{j-1} \sum_{i=0}^{l-2} \left( \prod_{i=j+1}^{l-1} \beta^i \right) r^i$$

$$= (I - \alpha^{j-1} A^*) (x^{j-1} - x) + \alpha^{j-1} A^* e$$

$$+ \alpha^{j-1} \sum_{i=0}^{l-2} \left( \prod_{i=j+1}^{l-1} \beta^i \right) (A^* (x - x^i) + A^* e). \quad (61)$$

Now, restricting (61) to $T \cup T^j$, taking the norm, and applying the triangle inequality yields (58).

□

We now use these two lemmas along with the technical lemmas from above to prove Theorem 2.

Proof of Thm. 2 (CGIHT): From Lemma 4, we can immediately see that

$$\| x^j - x \| \leq \sqrt{3} \| (w^{j-1} - x)_{T \cup T^j} \|. \quad (62)$$
Using notation from [19] updated for the sparsity in levels setting, we let

\[ \epsilon_a = \frac{1}{1 - L_k}, \]  
\[ \epsilon_\beta = \frac{(L_k + U_k)(1 + U_k)}{(1 - L_k)^2}, \]  
\[ \epsilon_\lambda = \frac{L_{3k} + U_{3k}}{1 - L_k}. \]  

Considering the right hand side of (62), we apply Lemma 5 to each term from Lemma 7, so that

\[ \|x' - x\|_2 \leq \sqrt{3}\|e_{l-1}^i - x\|_2 \]
\[ \leq \sqrt{3}\epsilon_a\|x^{l-1} - x\|_2 \]
\[ + \sqrt{3}\epsilon_\alpha\sqrt{1 + U_{2k}}\|e\|_2 \]
\[ + \sqrt{3}\epsilon_\alpha(1 + U_{3k})\sum_{j=0}^{l-2}\left(\epsilon_\beta^{l-j-1}\|x - x'\|_2\right) \]
\[ + \sqrt{3}\epsilon_\alpha\sqrt{1 + U_{2k}}\|e\|_2\sum_{j=0}^{l-2}\epsilon_\beta^{l-j-1}. \]  

Rearranging, these terms and re-indexing the sums provides the bound

\[ \|x' - x\|_2 \leq \sqrt{3}\epsilon_\lambda\|x^{l-1} - x\|_2 \]
\[ + \sqrt{3}\epsilon_\alpha\sum_{j=1}^{l-1}\left(\epsilon_\beta^{j}\|x^{l-j-1} - x\|_2\right) \]
\[ + \sqrt{3}\epsilon_\alpha\sqrt{1 + U_{2k}}\|e\|_2\sum_{j=0}^{l-1}\epsilon_\beta^j. \]  

Now, define \( c_0 = \|x - x^0\|_2 \) and \( c_1 = \sqrt{3}\epsilon_\lambda c_0 + \sqrt{3}\epsilon_\alpha\sqrt{1 + U_{2k}}\|e\|_2 \). Then (67) ensures that \( \|x^{l-1} - x\|_2 \leq c_1 \). Now, for each \( l \geq 2 \), define

\[ c_l = \sqrt{3}\epsilon_\lambda c_{l-1} + \sqrt{3}\epsilon_\alpha(1 + U_{3k})\sum_{j=1}^{l-1}\epsilon_\beta^j c_{l-j-1} \]
\[ + \sqrt{3}\epsilon_\alpha\sqrt{1 + U_{2k}}\|e\|_2\sum_{j=0}^{l-1}\epsilon_\beta^j. \]  

Then, (67) guarantees that \( \|x' - x\|_2 \leq c_l \) for all \( l \geq 2 \). From (68), we can extract a three term recurrence relation by rearranging the expression \( c_l - \epsilon_\beta c_{l-1} \) to obtain,

\[ c_l = \tau_1 c_{l-1} + \tau_2 c_{l-2} + \xi\|e\|_2 \]
\[ \tau_1 = \sqrt{3}\epsilon_\alpha + \epsilon_\beta, \]
\[ \tau_2 = \sqrt{3}(\epsilon_\alpha(1 + U_{3k}) - \epsilon_\lambda c_0)\epsilon_\beta, \]
\[ \xi = \sqrt{3}\epsilon_\alpha\sqrt{1 + U_{2k}}. \]  

Let \( \mu = \frac{1}{2}(\tau_1 + \sqrt{\tau_1^2 + 4\tau_2}). \) Since \( \epsilon_\beta > 0 \), then \( \sqrt{3}\epsilon_\lambda < \tau_1 < \mu \), so that \( c_1 \leq \mu c_0 + \xi\|e\|_2 \). Then, using Lemma 6 with \( \eta = \xi\|e\|_2 \), if \( \tau_1 + \tau_2 < 1 \), we have \( \mu < 1 \) and

\[ c_l \leq \mu^l c_0 + \xi\|e\|_2\sum_{j=0}^{l-1}\mu^j. \]  

When \( \mu < 1 \), we bound the finite sum of positive terms by its limit so that

\[ c_l \leq \mu^l c_0 + \frac{\xi}{1 - \mu}\|e\|_2. \]  

Finally, we determine a sufficient condition to ensure \( \tau_1 + \tau_2 < 1 \). Note that \( L_k \leq L_{3k} \) and \( U_k \leq U_{3k} \) so that \( U_k + L_k < U_{3k} + L_{3k} \) and \( 1 - L_{3k} \leq 1 - L_k \). Therefore, from (64)-(65) and (70), we have

\[ \tau_1 = \sqrt{3}\frac{U_{3k} + L_{3k}}{1 - L_k} + \frac{(U_k + L_k)(1 + U_k)}{(1 - L_k)^2} \leq \frac{U_{3k} + L_{3k}}{1 - L_k} \left( 1 + \sqrt{3} - \sqrt{3}L_k + U_k \right). \]  

Similarly, since

\[ \epsilon_a(1 + U_{3k}) - \epsilon_\lambda = \frac{1 - L_{3k}}{1 - L_k} < 1, \]  

we can bound (71) by

\[ \tau_2 \leq \frac{U_{3k} + L_{3k}}{1 - L_k} \left( \sqrt{3} + \sqrt{3}U_k \right). \]  

Thus, whenever

\[ \left( \frac{U_{3k} + L_{3k}}{1 - L_k} \left( 1 + 2\sqrt{3} - \sqrt{3}L_k + (1 + \sqrt{3})U_k \right) < 1, \]

we must have \( \tau_1 + \tau_2 < 1 \) and therefore (74) holds. Since we know \( \|x' - x\|_2 \leq c_l \) and \( c_0 = \|x^0 - x\|_2 \), we have the desired result

\[ \|x' - x\|_2 \leq \mu^l \|x^0 - x\|_2 + \frac{\xi}{1 - \mu}\|e\|_2. \]  

\[ \square \]

**ACKNOWLEDGMENT**

The author thanks the associate editor and five anonymous referees for their quality feedback that improved the article.

**REFERENCES**

[1] S. Kelly and R. Torres, “Guido weiss: From immigrant boy to internationally renowned mathematician,” J. Geometric Anal., vol. 31, pp. 9146–9179, 2021.
[2] D. L. Donoho, “Compressed sensing,” IEEE Trans. Inf. Theory, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[3] E. J. Candès, “Compressive sampling,” in Proc. Int. Congr. Mathematicians, Eur. Math. Soc., 2006, vol. 3, pp. 1433–1452.
[4] R. Gribonval and M. Nielsen, “Sparse representations in unions of bases,” IEEE Trans. Inf. Theory, vol. 49, no. 12, pp. 3320–3325, Dec. 2003.
[5] J. A. Tropp, “Greed is good: Algorithmic results for sparse approximation,” IEEE Trans. Inf. Theory, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.
[6] B. K. Natarajan, “Sparse approximate solutions to linear systems,” *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, 1995.

[7] D. L. Donoho, “For most large underdetermined systems of equations, the minimal ℓ1-norm near-solution approximates the sparsest near-solution,” *Commun. Pure Appl. Math.*, vol. 59, no. 7, pp. 907–934, 2006.

[8] D. L. Donoho and J. Tanner, “Sparse nonnegative solution of underdetermined linear equations by linear programming,” *Proc. Nat. Acad. Sci. USA*, vol. 102, no. 27, pp. 9446–9451, 2005.

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

[10] E. J. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information,” *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[11] J. A. Tropp and A. C. Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit,” *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[12] D. Needell and R. Vershynin, “Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit,” *Found. Comput. Math.*, vol. 9, no. 3, pp. 317–334, 2009.

[13] T. Blumensath and M. E. Davies, “Iterative hard thresholding for sparse approximations,” *J. Fourier Anal. Appl.*, vol. 14, no. 5/6, pp. 629–654, 2008.

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

[15] T. Blumensath and M. E. Davies, “Normalised iterative hard thresholding: guaranteed stability and performance,” *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 298–309, Apr. 2010.

[16] D. Needell and J. Tropp, “CoSaMP: Iterative signal recovery from incomplete and inaccurate samples,” *Appl. Comput. Harmon. Anal.*, vol. 26, no. 3, pp. 301–321, 2009.

[17] W. Dai and O. Milenkovic, “Subspace pursuit for compressive sensing signal reconstruction,” *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2230–2249, May 2009.

[18] S. Foucart, “Hard thresholding pursuit: An algorithm for compressive sensing,” *SIAM J. Numer. Anal.*, vol. 49, no. 6, pp. 2543–2563, 2011.

[19] J. D. Blanchard, J. Tanner, and K. Wei, “Conjugate gradient iterative hard thresholding for compressed sensing and matrix completion,” *Inf. Inference: J. IMA*, vol. 4, no. 4, pp. 289–327, Dec. 2015.

[20] D. L. Donoho, A. Maleki, and A. Montanari, “Message-passing algorithms for compressed sensing,” *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 45, pp. 18914–18919, 2009.

[21] Y. Lu and M. Do, “A theory for sampling signals from a union of subspaces,” *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2334–2345, Jun. 2008.

[22] T. Blumensath and M. E. Davies, “Sampling theorems for signals from the union of finite-dimensional linear subspaces,” *IEEE Trans. Inf. Theory*, vol. 55, no. 4, pp. 1872–1882, Apr. 2009.

[23] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde, “Model-based compressive sensing,” *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1982–2001, Apr. 2010.

[24] C. Hegde, P. Indyk, and L. Schmidt, “Approximation algorithms for model-based compressive sensing,” *IEEE Trans. Inf. Theory*, vol. 61, no. 9, pp. 5129–5147, Sep. 2015.

[25] B. Adcock, A. C. Hansen, C. Poon, and B. Roman, “Breaking the isometry constraint: A new theory for compressed sensing,” *Forum Math.*, vol. 5, 2017, Art. no. E4.

[26] B. Adcock, S. Brugiapaglia, and M. King-Roskamp, “The benefits of acting locally: Reconstruction algorithms for sparse in levels signals with stable and robust recovery guarantees,” *IEEE Trans. Signal Process.*, vol. 69, pp. 3160–3175, 2021.

[27] B. Adcock, S. Brugiapaglia, and M. King-Roskamp, “Do log factors matter? On optimal wavelet approximation and the foundations of compressed sensing,” *Found. Comput. Math.*, vol. 22, no. 1, pp. 99–159, 2022.

[28] A. Bastounis and A. C. Hansen, “On the absence of uniform recovery in many real-world applications of compressed sensing and the restricted isometry property and nullspace property in levels,” *SIAM J. Imag. Sci.*, vol. 10, no. 1, pp. 335–371, 2017.

[29] C. Li and B. Adcock, “Compressed sensing with local structure: Uniform recovery guarantees for the sparsity in levels class,” *Appl. Comput. Harmon. Anal.*, vol. 46, no. 3, pp. 453–477, 2019.

[30] P. Sprechmann, I. Ramírez, G. Sapiro, and Y. Eldar, “Collaborative hierarchical sparse modeling,” in *Proc. IEEE 44th Ann. Conf. Inf. Sci. Syst.*, 2010, pp. 1–6.

[31] G. Wunder, I. Roth, R. Fritschek, and J. Eisert, “HiHTP: A custom-tailored hierarchical sparse detector for massive MTC,” in *Proc. 51st Asilomar Conf. Signals, Syst., Comput.*, 2017, pp. 1929–1934.

[32] A. Finth, B. Groll, I. Roth, J. Eisert, and G. Wunder, “Hierarchical isometry properties of hierarchical measurements,” *Appl. Comput. Harmon. Anal.*, vol. 58, pp. 27–49, 2022.

[33] P. Breaney and J. Huang, “Penalized methods for bi-level variable selection,” *Statist. Interface*, vol. 2, no. 3, pp. 369–380, 2009.

[34] S. Xiang, T. Yang, and J. Ye, “Simultaneous feature and feature group selection through hard thresholding,” in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2014, pp. 532–541.

[35] C. Hegde, “Bilevel feature selection in nearly-linear time,” in *Proc. IEEE Stat. Signal Process. Workshop*, 2016, pp. 1–5.

[36] C. Boyer, J. Bigot, and P. Weiss, “Compressed sensing with structured sparsity and structured acquisition,” *Appl. Comput. Harmon. Anal.*, vol. 46, no. 2, pp. 312–350, 2019.

[37] J. S. Donato and H. W. Levinson, “Structured iterative hard thresholding with on- and off-grid applications,” *Linear Algebra Appl.*, vol. 638, pp. 46–79, 2022.

[38] J. D. Blanchard and J. Tanner, “Performance comparisons of greedy algorithms for compressed sensing,” *Numer. Linear Algebra Appl.*, vol. 22, no. 2, pp. 254–282, 2015.

[39] J. D. Blanchard, C. Cartis, and J. Tanner, “Compressed sensing: How sharp is the restricted isometry property?,” *SIAM Rev.*, vol. 53, no. 1, pp. 105–125, 2011.

[40] J. D. Blanchard, M. Cermak, D. Hanle, and Y. Jing, “Greedy algorithms for joint sparse recovery,” *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1694–1704, Apr. 2014.

[41] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. New York, NY, USA: Birkhäuser, 2013.

[42] A. Greenbaum, *Iterative Methods for Solving Linear Systems*. Philadelphia, PA, USA: SIAM, 1997.

[43] J. D. Blanchard and J. Tanner, “GPU accelerated greedy algorithms for compressed sensing,” *Math. Prog. Comput.*, vol. 5, no. 3, pp. 267–304, 2013.

[44] A. Maleki and D. Donoho, “Optimal locally iterative reconstruction algorithms for compressed sensing,” *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 330–341, Apr. 2010.

[45] J. D. Blanchard, J. Tanner, and K. Wei, “Conjugate gradient iterative hard thresholding: Observed noise stability for compressed sensing,” *IEEE Trans. Signal Process.*, vol. 63, no. 2, pp. 528–537, Jan. 2015.