Partial Hard Thresholding

Prateek Jain, Ambuj Tewari, and Inderjit S. Dhillon

***Abstract*—We study iterative algorithms for compressed sensing that have an "orthogonalization" step at each iteration to keep the residual orthogonal to the span of those columns of the measurement matrix that have been selected so far. To unify the design and analysis of such algorithms, we propose a novel partial hard-thresholding (PHT) operator that is similar to the hard thresholding operator but restricts the amount by which the support set can change in one iteration. Using the PHT operator and its properties, we provide a general framework to prove support recovery results for iterative algorithms employing this operator as well as those employing the hard-thresholding operator. Next, based on the PHT operator, we propose a novel family of algorithms. At one end of our family of algorithms lie well-known hard thresholding algorithms ITI [1] and HTP [2], whereas at the other end, we get a novel algorithm that we call Orthogonal Matching Pursuit with Replacement (OMPR). Like the classic greedy algorithm OMP, OMPR too adds exactly one coordinate to the support of the iterate at each iteration based on the correlation with the current residual. However, unlike OMP, OMPR also removes one coordinate from the support. This simple change allows us to prove that OMPR has the best known guarantees for sparse recovery in terms of the Restricted Isometry Property (RIP), a condition on the measurement matrix. In contrast, OMP is known to have very weak performance guarantees under RIP.**

Finally, we show that most of the existing "orthogonalized" iterative algorithms such as CoSaMP, Subspace Pursuit, OMP, can be expressed using the PHT operator. As a pleasing consequence of our novel and generic results for the PHT operator, we provide the tightest known RIP analysis of all of the above mentioned iterative algorithms: CoSaMP, Subspace Pursuit, and OMP.

***Index Terms*—compressed sensing, sparse recovery, restricted isometry property, iterative thresholding algorithms**

I. Introduction

Consider the compressed sensing setting [3], [4] where we wish to efficiently recover a *sparse* vector $x^\star \in \mathbb{R}^n$ using a small number m of linear measurements $b = Ax^\star \in \mathbb{R}^m$. The measurement matrix $A \in \mathbb{R}^{m\times n}$ is often chosen from an appropriate random matrix ensemble. Such a choice ensures that there are efficient recovery algorithms that will, with high probability, recover any k-sparse vector using just $O(k\log(n/k))$ measurements. Candes and Tao [3] isolated a key property of the matrix A, called the Restricted Isometry Property (RIP), and proved that, as long as A satisfies RIP, the true sparse vector $x^\star$ can be obtained by solving an ℓ_1-optimization problem,

$$\min \ \|x\|_1 \text{ s.t. } Ax = b \ .$$

The above problem can be easily formulated as a linear program and can therefore be solved efficiently. We recall that a matrix A is said to satisfy RIP of order k if there is some $\delta_k \in [0,1)$ such that,

$$\forall x \text{ s.t. } \|x\|_0 \le k, \ (1-\delta_k)\|x\|^2 \le \|Ax\|^2 \le (1+\delta_k)\|x\|^2 \ .$$

Here $\|x\|$ is the standard Euclidean (or ℓ_2) norm of x, and $\|x\|_0$ is the so-called "ℓ_0-norm": the size of $\mathrm{supp}(x)$, the support of x.

Several random matrix ensembles, such as the Gaussian and the symmetric Bernoulli ensembles, are known to satisfy the condition $\delta_{ck} < \theta$ with high probability provided one chooses $m = O(\frac{ck}{\theta^2}\log\frac{n}{k})$ measurements. It has been shown [5] that ℓ_1-minimization recovers any k-sparse vector x from measurements Ax as soon as A satisfies $\delta_{2k} < \sqrt{2}-1 \approx 0.414$. This condition has since been improved to $\delta_{2k} < 0.493$ [6].

Even though ℓ_1-minimization can be performed efficiently using convex optimization techniques, it requires substantial computation expense in large scale applications [7]: for example, when n is in the millions. If the sparsity level k is much smaller than n, iterative methods that solve optimization problems involving k, instead of n, variables at each iteration, become attractive alternatives to ℓ_1-minimization. A classic iterative method is Orthogonal Matching Pursuit (OMP) [8], [9] that greedily chooses elements to add to the support. It is a natural, easy to implement, and fast method but it unfortunately lacks strong theoretical guarantees. Indeed, it is known that, if OMP is only run for k iterations, it cannot recover all k-sparse vectors assuming an RIP condition of the form $\delta_{2k} < \theta$ [10], [11]. However, Zhang [12] has shown that OMP, if run for $30k$ iterations, will recover the true sparse vector when $\delta_{31k} < 1/3$. The support size $31k$ in the RIP condition makes it a significantly more restrictive condition that the ones requires by other methods like ℓ_1-minimization.

Several other iterative methods, with better guarantees, have been proposed in the literature. A partial list includes Iterative Soft Thresholding (IST) [1], Iterative Hard Thresholding (IHT) [13], Compressive Sampling Matching Pursuit (CoSaMP) [14], Subspace Pursuit (SP) [15], Iterative Thresholding with Inversion (ITI) [16], and Hard Thresholding Pursuit (HTP) [2]. Following Maleki and Donoho [1], we can classify these iterative thresholding algorithms into two major families: one-stage and two-stage algorithms. One-stage algorithms such as IHT, ITI, and HTP, determine the choice of the next support and then usually solve a least squares problem on the updated support. One-stage methods always set the support to have size k, where k is the target sparsity level. On the other hand,

Prateek Jain is with Microsoft Research Bangalore, India.

Ambuj Tewari is with the Department of Statistics and Department of Electrical Engineering & Computer Science, University of Michigan, Ann Arbor.

Inderjit S. Dhillon is with the Department of Computer Science, University of Texas at Austin.

Manuscript received Month DD, YYYY; revised Month DD, YYYY.

This paper was presented in part at NIPS 2011.

two-stage algorithms like CoSaMP and SP, first *enlarge* the support, solve a least squares problem on it, and subsequently *reduce* the support back to the desired size k. A second least squares problem is then solved on the reduced support. These algorithms typically enlarge and reduce the support by k or $2k$ elements. An exception is the two-stage algorithm FoBa [17] that adds and removes single elements from the support.

Another criterion for classification of iterative methods for sparse recovery is whether or not they keep the current residual orthogonal to the span of the columns columns corresponding to the current support. Such algorithms are called "fully corrective" (or "totally corrective") in the boosting literature [18]. For general error criteria, fully corrective methods keep each iterate optimal (in terms of the underlying error criterion) over the current support set. When the objective function is the squared error, fully corrective methods solve a least squares problem over variables in the current support. Most of the existing methods are indeed fully corrective: for example, OMP, FoBa, HTP, CoSaMP, and SP. On the other hand, Matching Pursuit is not fully-corrective. In this paper, we unify both the presentation as well as the analysis of fully corrective methods using two operators: a) a novel operator that we call Partial Hard Thresholding (PHT), and b) the standard hard thresholding operator (HT). We show that most of the existing fully corrective methods can be obtained by combining the PHT and HT operators appropriately. Moreover, our general analysis for PHT and HT operators enables us to provide recovery guarantees under RIP requirements that are the best known so far. In particular, we are able to improve CoSaMP's RIP condition for sparse recovery to $\delta_{4k} < 0.39$. For SP, the condition improves to $\delta_{3k} < 0.39$. Moreover, we also improve OMP's recovery condition to $\delta_{3k} < 0.2$.

Next, we propose and analyze a *novel* family of one-stage iterative thresholding algorithms that we call Partial Hard Thresholding (PHT). The family is parameterized by a positive integer $\ell \leq k$. We denote a particular member of the family by PHT(ℓ). At the extreme value $\ell = k$, we recover the algorithm that is Maleki's ITI or Foucart's HTP. At the other extreme $\ell = 1$, we get a novel algorithm that we call Orthogonal Matching Pursuit with Replacement (OMPR). The name reflects the fact that OMPR can be thought of as a simple modification of OMP: instead of simply *adding* an element to the existing support, it *replaces* an existing support element with a new one. Surprisingly, this simple change allows us to give sparse recovery guarantees under the condition $\delta_{2k} < 0.499$. At present, this is the best δ_{2k} based RIP condition under which *any* method, including ℓ_1-minimization, is known to provably perform sparse recovery. We note that RIP based analysis is not the only theoretical tool to understand the behavior of sparse recovery methods. Practical performance on some problems might be better explained by phase transition calculations under various random ensembles for the signal and measurements (see, for example, [7], [19]–[21]).

An added advantage of OMPR, unlike many iterative methods, is that no careful tuning of the step-size parameter is required under noisy settings or even when RIP does not hold. The default step-size of 1 is always guaranteed to converge to a local optimum of the objective function.

A. Preliminaries

For a positive integer n, we denote $\{1, \ldots, n\}$ by $[n]$. For a size k set $I \subseteq [n]$, $\bar{I}$ denotes its set complement relative to $[n]$. For a matrix $A \in \mathbb{R}^{m \times n}$, A_I denotes the $m \times k$ submatrix of A corresponding to columns indexed by I. Similarly, for a vector $x \in \mathbb{R}^n$, $x_I \in \mathbb{R}^k$ is the subvector corresponding to entries of x indexed by I. We use the short hand $A \backslash b$ to denote the solution of an (overdetermined) least squares problem

$$A \backslash b := \operatorname*{argmin}_x \ \|Ax - b\|^2 \ .$$

II. Partial Hard Thresholding

To motivate Partial Hard Thresholding, let us look at the one of simplest optimization algorithms, namely Projected Gradient Descent. It generates iterates as follows

$$z^{t+1} \leftarrow x^t - \eta \nabla f(x^t) \ , \tag{1}$$

$$x^{t+1} \leftarrow \operatorname*{argmin}_{\|x\|_0 \leq k} \ \|x - z^{t+1}\| \ , \tag{2}$$

where $\eta > 0$ is a step-size parameter and the function f in our case is

$$f(x) = \tfrac{1}{2}\|Ax - b\|^2 \ .$$

The projection onto the set of k-sparse vectors is the so called *hard thresholding operator*

$$H_k(z) := \operatorname*{argmin}_{\|x\|_0 \leq k} \ \|x - z\| \ .$$

Even though the set of k-sparse vectors in $\mathbb{R}^n$ is non-convex, the projection of a vector x onto it can be computed efficiently: simply retain the top k entries of x in absolute value (and convert the rest to zeros).

In terms of the hard thresholding operator, projected gradient descent for the squared error objective can be written as

$$z^{t+1} \leftarrow x^t - \eta A^T(Ax^t - b) \ ,$$
$$x^{t+1} \leftarrow H_k\left(z^{t+1}\right) \ .$$

This simple algorithm is called Iterative Hard Thresholding (IHT). After the projection step, we may additionally solve a least squares problem on the support of the new iterate giving us the following algorithm.

$$z^{t+1} \leftarrow x^t - \eta A^T(Ax^t - b) \ ,$$
$$y^{t+1} \leftarrow H_k\left(z^{t+1}\right) \ ,$$
$$I_{t+1} \leftarrow \operatorname{supp}(y^{t+1}) \ ,$$
$$x^{t+1}_{I_{t+1}} \leftarrow A_{I_{t+1}} \backslash b, \ x^{t+1}_{\bar{I}_{t+1}} \leftarrow \mathbf{0} \ .$$

The last step is also sometimes referred to as an *orthogonalization* step since it renders the residual orthogonal to the columns of $A_{I_{t+1}}$. This fact will be used crucially in our analysis. The IHT algorithm with an additional least squares step has been called Iterative Thresholding with Inversion (ITI) by Maleki [16] and Hard Thresholding Pursuit (HTP) by Foucart [2]. Our own preference is to call it IHT-Newton to emphasize its close relationship with IHT. The "Newton" suffix denotes the fact that, for the least squares objective, a full minimization is equivalent to a Newton step. Our analysis

will assume that the least squares problems, such as the one above, are solved exactly. In practice, of course, one would run a few steps of an iterative procedure such as conjugate gradient. Because these least squares problems are solved on a small number of variables, the RIP condition ensures that one is solving well-conditioned least squares problems which means that these iterative least square solvers converge linearly. We do not pursue the error analysis caused by running an iterative solver for a finite number of steps because such analyses can be found elsewhere (see, for example, [14, Section 5]).

To generalize IHT-Newton, we need to generalize the hard thresholding operator. A natural generalization is obtained by further constraining the set onto which we project the gradient. In addition to the sparsity contraint, we add the constraint that the support should not loose more than ℓ elements relative to the previous support. Thus, we define the partial hard thresholding operator

$$PHT_k\left(z; I, \ell\right) := \underset{\|x\|_0 \leq k,\ |I \setminus \operatorname{supp}(x)| \leq \ell}{\operatorname{argmin}} \|x - z\| \ . \quad (3)$$

Compared to hard thresholding, the PHT operator takes two additional arguments: an upper bound ℓ on the support change, and a set I relative to which the change is measured. The following lemma guarantees that the PHT operator can be computed efficiently.

Lemma 1: Fix a set $I \subseteq [n]$ of size $k' \geq 0$, a vector $z \in \mathbb{R}^n$, and a positive integer $\ell \leq n - k'$. Then $y = PHT_k\left(z; I, \ell\right)$ can be computed using the following sequence of operations:

$$\begin{aligned} \text{top} &\leftarrow \text{indices of largest}^1\ k + \ell - k' \text{ elements of } z_{\bar{I}}\ , \\ \text{bot} &\leftarrow \text{indices of smallest } \ell \text{ elements of } z_I, \\ J &= \operatorname{supp}(H_{k-k'+\ell}\left(z_{\text{bot}\cup\text{top}}\right)) \cup (I \setminus \text{bot})), \\ y_J &= z_J, y_{\bar{J}} = \mathbf{0}. \end{aligned}$$

Algorithm 1 PHT(ℓ)

Initialize x^1 s.t. $|\operatorname{supp}(x^1)| = k$
$I_1 \leftarrow \operatorname{supp}(x^1)$
for $t = 1, 2, \ldots$ **do**
  /* Gradient Descent */
  $z^{t+1} \leftarrow x^t - \eta A^T(Ax^t - b)$

  /* Partial Hard Thresholding */
  $y^{t+1} \leftarrow PHT_k\left(z^{t+1}; I_t, \ell\right)$

  /* Solve a least squares problem */
  $I_{t+1} \leftarrow \operatorname{supp}(y^{t+1})$
  $x^{t+1}_{I_{t+1}} \leftarrow A_{I_{t+1}} \backslash b,\ x^{t+1}_{\bar{I}_{t+1}} \leftarrow \mathbf{0}$
end for

Our new algorithm family PHT(ℓ) (see Algorithm 1) is obtained simply by replacing the hard thresholding operator in IHT-Newton by the more general PHT operator. Since the PHT operator becomes the hard thresholding operator for the choice $\ell = k$ (and any choice for the set I), it is clear that the PHT(k) algorithm is nothing but IHT-Newton (or ITI or HTP). Choices for ℓ strictly smaller than k yield novel algorithms. In particular, the choice $\ell = 1$ at the other extreme yields a particularly interesting algorithm. It turns out that PHT(1) is closely connected to the classic OMP algorithm. In view of this connection, we give it a special name: Orthogonal Matching Pursuit with Replacement (OMPR). We now expand on the OMPR algorithm and its connection to OMP.

[1]Here, "largest" and "smallest" refer to sorting elements according to their absolute values.

Algorithm 2 OMP

$x^1 \leftarrow \mathbf{0}$
$I_1 \leftarrow \emptyset$
for $t = 1, 2, \ldots$ **do**
  /* Gradient Descent */
  $z^{t+1} \leftarrow x^t - \eta A^T(Ax^t - b)$

  /* Compute $PHT_{t+1}\left(z^{t+1}; I_t, 1\right)$ */
  $j_{t+1} \leftarrow \operatorname{argmax}_{j \notin I_t}\ |z^{t+1}_j|$
  $J_{t+1} \leftarrow I_t \cup \{j_{t+1}\}$
  $y^{t+1} \leftarrow H_{t+1}\left(z^{t+1}_{J_{t+1}}\right)$

  /* Solve a least squares problem */
  $I_{t+1} \leftarrow \operatorname{supp}(y^{t+1})$
  $x^{t+1}_{I_{t+1}} \leftarrow A_{I_{t+1}} \backslash b,\ x^{t+1}_{\bar{I}_{t+1}} \leftarrow \mathbf{0}$
end for

A. Orthogonal Matching Pursuit with Replacement

The classic OMP algorithm is based on a very simple idea. At each step, find a new column of A that is maximally correlated with the current residual $Ax^t - b$. Then add that column's index to the current support followed by a least squares step on the expanded support. In symbols,

$$\begin{aligned} j_{t+1} &\leftarrow \underset{j \notin I_t}{\operatorname{argmax}}\ A_j^T(Ax^t - b)\ , \\ I_{t+1} &\leftarrow I_t \cup \{j_{t+1}\}\ , \\ x^{t+1}_{I_{t+1}} &\leftarrow A_{I_{t+1}} \backslash b,\ x^{t+1}_{\bar{I}_{t+1}} \leftarrow \mathbf{0}\ . \end{aligned}$$

Define the gradient descent iterate

$$z^{t+1} \leftarrow x^t - \eta A^T(Ax^t - b)\ .$$

Because x^t has zero entries outside the support I_t, we can rewrite OMP as Algorithm 2 which highlights its connections to the PHT family.

Now consider the PHT(1) or OMPR algorithm (see Algorithm 3). The only difference between it and OMP is that, after including the new element into the support, OMPR removes an element using a hard thresholding operation. As a result, the new element *replaces* an existing element of the support (in general). The least squares problem in OMPR is always solved on a support of size k. In contrast, OMP can increase the support size beyond k if it is run for more than k iterations.

OMP is not known to enjoy strong RIP based sparse recovery guarantees. In fact, Mo and Shen [11] show that OMP can fail to recover a sparse vector from linear measurement if it is only run for k iterations. This is despite the fact that A has an RIP constant that be made arbitraily small with k. We

Algorithm 3 OMPR

Initialize x^1 s.t. $|\operatorname{supp}(x^1)| = k$
$I_1 \leftarrow \operatorname{supp}(x^1)$
for $t = 1, 2, \ldots$ **do**
  /* Gradient Descent */
  $z^{t+1} \leftarrow x^t - \eta A^T(Ax^t - b)$

  /* Compute $PHT_k\left(z^{t+1}; I_t, 1\right)$ */
  $j_{t+1} \leftarrow \operatorname{argmax}_{j \notin I_t} |z_j^{t+1}|$
  $J_{t+1} \leftarrow I_t \cup \{j_{t+1}\}$
  $y^{t+1} \leftarrow H_k\left(z^{t+1}_{J_{t+1}}\right)$

  /* Solve a least squares problem */
  $I_{t+1} \leftarrow \operatorname{supp}(y^{t+1})$
  $x^{t+1}_{I_{t+1}} \leftarrow A_{I_{t+1}} \backslash b,\ x^{t+1}_{\bar{I}_{t+1}} \leftarrow \mathbf{0}$
end for

show below how OMPR is able to succeed on the example from [11].

1) A Bad Case for OMP: Let the measurement matrix A be given by

$$A = \begin{bmatrix} & & & \frac{1}{k} \\ & I_k & & \vdots \\ & & & \frac{1}{k} \\ 0 & \ldots & 0 & \sqrt{\frac{k-1}{k}} \end{bmatrix} .$$

Let the true sparse vector be $x^\star = (1, \ldots, 1, 0)^T$. It is easy to verify that A's RIP constant of order $k+1$ satisfies $\delta_{k+1} < \frac{1}{\sqrt{k}}$.

At the first iteration, the residual is the measurement vector $b = Ax^\star$ itself. The correlation $A_j^T Ax^\star$ is 1 for all $j \in [k+1]$. Thus, OMP can select an incorrect index at the first iteration leading to an incorrect solution at the end of k iterations. We now show that, on the same example, OMPR is indeed able to recover $x^\star$. Suppose, in the first k steps, OMPR selects an incorrect support. By symmetry among the first k coordinates, we can assume without loss of generality that the incorrect support is $I_{k+1} = \{2, 3, \ldots, k+1\}$. Solving a least squares problem on I_{k+1} gives

$$x^{k+1} = (0, \tfrac{k^2-k}{k^2-k+1}, \ldots, \tfrac{k^2-k}{k^2-k+1}, \tfrac{k}{k^2-k+1})^T .$$

The gradient $A^T A(x^{k+1} - x^\star)$ becomes

$$\begin{bmatrix} & & & \frac{1}{k} \\ & I_k & & \vdots \\ & & & \frac{1}{k} \\ \frac{1}{k} & \ldots & \frac{1}{k} & 1 \end{bmatrix} \begin{bmatrix} -1 \\ \frac{-1}{k^2-k+1} \\ \vdots \\ \frac{-1}{k^2-k+1} \\ \frac{k}{k^2-k+1} \end{bmatrix} = \begin{bmatrix} \frac{-k^2+k}{k^2-k+1} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} .$$

Note that, as expected, the gradient has zero entries on the support I_{k+1} because of the least squares step. Thus, at iteration number $k+1$ with step-size $\eta = 1$, we have

$$z^{k+2} = (\tfrac{k^2-k}{k^2-k+1}, \ldots, \tfrac{k^2-k}{k^2-k+1}, \tfrac{k}{k^2-k+1})^T .$$

OMPR then selects $j_{k+2} = 1$ leading to $J_{k+2} = I_{k+1} \cup \{j_{k+2}\} = [k+1]$. Hard thresholding $z^{k+2}_{J_{k+2}}$ now drops the last entry giving

$$y^{k+2} = (\tfrac{k^2-k}{k^2-k+1}, \ldots, \tfrac{k^2-k}{k^2-k+1}, 0)^T .$$

This leads to I_{k+2} becoming the correct support $[k]$. As a result $x^{k+2} = x^\star$ showing that OMPR is able to recover the true sparse vector by solving least squares problems of size at most k.

In the above discussion we assumed that OMP is run only for k iterations. OMP does have an RIP analysis assuming it is run for more than k iterations. To the best of our knowledge, the best RIP analysis of OMP is the one by Zhang [12] where he gives the sufficient condition $\delta_{31k} < 1/3$ for OMP run for $30k$ iterations. We now provide an RIP analysis of PHT(ℓ) that will show that OMPR requires a much weaker RIP condition compared to OMP's.

B. RIP Based Guarantees for PHT

We now present RIP based guarantees for the PHT family of algorithms (for any general ℓ).

We first present the guarantees for the entire PHT family, when $b = Ax^\star$.

Theorem 2: Suppose the vector $x^\star \in \mathbb{R}^n$ is k-sparse. Then PHT(ℓ) converges to an ϵ approximation solution (i.e. $\frac{1}{2}\|Ax - b\|^2 \leq \epsilon$) from measurements $b = Ax^\star$ in $O(\frac{k}{\ell}\log(k/\epsilon))$ iterations provided we choose a step size η that satisfies $\eta(1 + \delta_{2\ell}) < 1$ and $\eta(1 - \delta_{2k}) > 1/2$.

The proof of Theorem 2 can be found in Section VII-A. Note that in the theorem above we guarantee recovery in terms of the least squares criterion, i.e., $\frac{1}{2}\|Ax - Ax^\star\|^2 \leq \epsilon$. However, when $\delta_{2k} < 1$, this immediately implies that $\|x - x^\star\| \leq 2\epsilon/(1 - \delta_{2k})$.

Note that the number of iterations required increase as ℓ decreases. On the other hand, each individual iteration will tend to be faster for smaller ℓ. Also note that the condition under which PHT(ℓ) recovers sparse vectors becomes more restrictive as ℓ increases. However, this could be an artifact of our analysis, as in experiments, we do not see any degradation in recovery ability as ℓ is increased. We do not report any experimental results in this paper but the interested reader can find some experimental results, especially about OMPR, in our earlier work [22].

As mentioned above, the OMPR algorithm of the previous section is simply PHT(1). Hence, the recovery guarantee for OMPR is a direct corollary of Theorem 2.

Corollary 3: Suppose the vector $x^\star \in \mathbb{R}^n$ is k-sparse and the matrix A satisfies $\delta_{2k} < 0.499$ and $\delta_2 < 0.002$. Then OMPR converges to an ϵ approximate solution (i.e., $\frac{1}{2}\|Ax - b\|^2 \leq \epsilon$) from measurements $b = Ax^\star$ in $O(k\log(k/\epsilon))$ iterations.

Beyond requiring $\delta_{2k} < 0.499$, there is an extra condition: namely $\delta_2 < 0.002$. This condition is very mild: for many random measurements, including Gaussian, with a scaling that keeps each column have a constant Euclidean norm, the inner product between any two columns is, with high probability, $O(1/\sqrt{m})$. Taking a union bound over $O(n^2)$ pairs of columns

tells us that $\delta_2 < 0.002$ holds as soon as $m > O(\log n)$ (note that there is no k here).

Similar to the extension of OMPR's analysis to the noisy case [22, Theorem 5], i.e., when $b = Ax^\star + e$, our analysis can be easily extended to handle noise. In particular, since our main lemma (Lemma 11) holds in the noisy case as well, one can show that PHT(ℓ) converges to a (C, ϵ) approximate solution (i.e., $\frac{1}{2}\|Ax - b\|^2 \leq \frac{C}{2}\|e\|^2 + \epsilon$) in $O(\frac{k}{\ell}\log((k + \|e\|^2)/\epsilon))$ iterations provided we choose a step size η that satisfies $\eta(1 + \delta_{2\ell}) < 1$ and $\eta(1 - \delta_{2k}) > 1/2$. Here $C > 1$ is a constant dependent only on $\delta_{2\ell}, \delta_{2k}$.

Among different δ_{2k} based conditions under which sparse recovery methods, including ℓ_1-minimization, are current known to work, OMPR's condition is the best (i.e., weakest). However, sufficient conditions for sparse recovery methods are often stated using RIP constants of a higher order than $2k$. It is, in general, not possible to compare a δ_{2k} condition with, say a δ_{3k} condition. Foucart [2], however, has suggested a *heuristic* to compare conditions of the form $\delta_{ck} < \theta$ with each other. The heuristic is based on the fact that, for various random matrix ensembles, it takes $m = O(\frac{ck}{\theta^2}\log\frac{n}{k})$ rows for the matrix A to satisfy $\delta_{ck} < \theta$. Thus, the smaller the ratio c/θ^2, the weaker the RIP condition becomes, heuristically speaking. From this perspective, OMPR's $\delta_{2k} < 0.499$ condition is currently the best known RIP condition leading to a c/θ^2 ratio of $2/0.499^2 \approx 8$.

III. Two-Stage Hard Thresholding

Maleki and Donoho [1] pointed out that popular sparse recovery algorithms such as CoSaMP and Subspace Pursuit can be understood as special cases of a general family that they called Two-Stage Thresholding (TST) algorithms. According to their empirically observed phase transitions, TST algorithms can have better sparse recovery properties compared to simpler, single stage algorithms. As the name suggests, in each iteration of these algorithms there are two thresholding steps. These thresholding steps sandwich a least squares step between them. Our Two-Stage family is essentially the same as Maleki and Donoho's TST except that we end the iteration by a second least squares or orthogonalization step. This yields the family depicted in Algorithm 4.

Note that Subspace Pursuit, as defined by its authors, does have the second least squares step. As for CoSaMP, the version with the second least squares step is mentioned in a section on "Other Variants" (Section A.2 in [14]) with the remark that "we can solve another least-squares problem in an effort to improve the final result". We therefore define our Two-Stage family with the second least squares step included.

It is interesting to note the similarities and differences between one stage algorithms, such as PHT(ℓ), and the Two-Stage family. The thresholding step in both kinds of algorithms can be written in terms of the partial hard thresholding operator and its special case, the hard thresholding operator. Moreover, small least square problems of sizes that are small multiples of k are solved in each iteration in either type of algorithms. These least squares problems are guaranteed to be well-conditioned if RIP of an appropriate order holds. The differences are equally obvious. Two-Stage algorithm solve two least squares problems per iteration. Further, the first least squares problem is of a larger size than the target sparsity level k unlike PHT(ℓ) where the size of the least squares problem is always k.

Algorithm 4 Two-Stage(ℓ)

repeat

/* Gradient Descent */
$z^{t+1} \leftarrow x^t - \eta A^T(Ax^t - b)$

/* First thresholding */
$y^{t+1} \leftarrow PHT_{k+\ell}\left(z^{t+1}; I_t, \ell\right)$

/* First least squares */
$J_{t+1} \leftarrow \mathrm{supp}(y^{t+1})$
$w^{t+1}_{J_{t+1}} \leftarrow A_{J_{t+1}}\backslash b,\ x^{t+1}_{\bar{J}_{t+1}} \leftarrow \mathbf{0}$

/* Second thresholding */
$v^{t+1} \leftarrow H_k\left(w^{t+1}\right)$

/* Second least squares */
$I_{t+1} \leftarrow \mathrm{supp}(v^{t+1})$
$x^{t+1}_{I_{t+1}} \leftarrow A_{I_{t+1}}\backslash b,\ x^{t+1}_{\bar{I}_{t+1}} \leftarrow \mathbf{0}$

until convergence

Our next result gives a general performance guarantee for the entire Two-Stage family under RIP.

Theorem 4: Suppose the vector $x^\star \in \mathbb{R}^n$ is k-sparse. Then the Two-stage Hard Thresholding algorithm with replacement size $\ell \geq k$ recovers $x^\star$ from measurements $b = Ax^\star$ in $O(\log k)$ iterations provided: $\delta_{2k+\ell} \leq 0.46$.

The proof of Theorem 4 can be found in Section VII-B.

Since CoSaMP (with an additional least squares step at the end of each iteration) is simply Two-Stage($2k$), we immediately get the following guarantee for CoSaMP.

Corollary 5: CoSaMP [14] recovers k-sparse $x^\star \in \mathbb{R}^n$ from measurements $b = Ax^\star$ provided $\delta_{4k} \leq 0.46$.

Similarly, we get the following guarantee for Subspace Pursuit which is the same as our Two-Stage(k) algorithm.

Corollary 6: Subspace Pursuit [15] recovers k-sparse $x^\star \in \mathbb{R}^n$ from measurements $b = Ax^\star$ provided $\delta_{3k} \leq 0.46$.

IV. Improved RIP Based Guarantees for OMP

As we discussed above in Section II-A, the classic OMP algorithm proceeds by iteratively adding elements to the support of the iterate. In some papers, the description of OMP implies that it is run only for k iterations. Unfortunately, it is known [10], [11] that OMP, when run only for k iterations cannot perform sparse recovery under RIP conditions of the form $\delta_{ck} < \theta$ under which PHT(ℓ), CoSaMP, and Subspace Pursuit are all known to work. Zhang [12] settled the question of whether OMP, when run for more than k iterations, can have optimal sparse recovery guarantees under RIP. He showed that running OMP for $30k$ iterations uniformly recovers a sparse vector from linear measurements provided the RIP constant of the measurement matrix satisfies $\delta_{31k} < 1/3$. Using the

proof techniques used in the analysis of PHT(ℓ), we are able to further improve his RIP condition.

Theorem 7: Let $x^* \in \{-1, 0, 1\}^n$ be a k-sparse vector. Let $b = Ax^*$ and $x^0 = 0^n$. Then, OMP recovers back optimal x^* given that following RIP condition is satisfied:

$$\delta_{4k} \leq 0.2 \text{ or } \delta_{5k} \leq 0.33.$$

The proof of Theorem 7 can be found in Section VII-C.

V. Proof Idea

Our proofs exploit a structural property of the gradient descent iterate

$$z^{t+1} = x^t - \eta A^T(Ax^t - b) \ .$$

Since x^t is obtained by a least squares step on the support set I_t, its residual is orthogonal to the column space of A_{I_t}. Hence we have the following structural decomposition for z^{t+1}.

$$\begin{aligned} z^{t+1}_{I_t} &= x^t_{I_t} \ , \\ z^{t+1}_{\bar{I}_t} &= -\eta A^T_{\bar{I}_t}(Ax^t - b) \ . \end{aligned} \tag{4}$$

During the partial hard thresholding step, elements move in and out of the support. For PHT(ℓ) and OMP, let us define

$$\begin{aligned} F_t &= I_{t+1} \backslash I_t && \text{(found)} \\ L_t &= I_t \backslash I_{t+1} && \text{(lost)} \\ R_t &= I_t \cap I_{t+1} \ . && \text{(retained)} \end{aligned}$$

For Two-Stage(ℓ), let us define

$$\begin{aligned} F_t &= J_{t+1} \backslash I_t && \text{(found)} \\ L_t &= I_t \backslash J_{t+1} && \text{(lost)} \\ R_t &= I_t \cap J_{t+1} \ . && \text{(retained)} \end{aligned}$$

Note that, for Two-Stage(ℓ), we have $I_t \subseteq J_{t+1}$ by definition and hence $L_t = \emptyset$, $R_t = I_t$.

Define the least squares objective

$$f(x) = \tfrac{1}{2}\|Ax - b\|^2 \ .$$

The goal of the analysis is to show that, for any of the algorithms PHT(ℓ), OMP, or Two-Stage(ℓ), the objective function difference $f(x^{t+1}) - f(x^t)$ is sufficiently negative.

Recall that, in OMP as well as both the families of algorithms, namely, PHT(ℓ), Two-Stage(ℓ), a key step is:

$$y^{t+1} = PHT_\alpha(z^{t+1}; I_t, \ell),$$

where $\ell = 1$ for OMP. Also, recall that $\alpha = k$ for PHT(ℓ), $\alpha = k + \ell$ for Two-Stage(ℓ), and $\alpha = t + 1$ for OMP.

Now,

$$\begin{aligned} f(y^{t+1}) - f(x^t) &= (y^{t+1} - x^t)^T A^T(Ax^t - b) \\ &\qquad + 1/2\|A(y^{t+1} - x^t)\|^2, \\ &\leq (y^{t+1} - x^t)^T A^T(Ax^t - b) \\ &\quad + \frac{(1 + \delta_{|F_t|+|L_t|})}{2}(\|y^{t+1}_{F_t}\|^2 + \|x^t_{L_t}\|^2). \end{aligned} \tag{5}$$

where the second inequality follows by using the fact that $y^{t+1}_{I_{t+1} \cap I_t} = x^t_{I_{t+1} \cap I_t}$ and using RIP of order $|F_t| + |L_t|$ (since $|\operatorname{supp}(y^{t+1} - x^t)| = |F_t \cup L_t| = |F_t| + |L_t|$).

Since $x^t_{I_t}$ is obtained using least squares,

$$A^T_{I_t}(Ax^t - b) = \mathbf{0}.$$

Thus, $A^T_{L_t}(Ax^t - b) = \mathbf{0}$, because $L_t \subseteq I_t$. Next, note that

$$y^{t+1}_{F_t} = -\eta A^T_{F_t}(Ax^t - b).$$

Hence,

$$\begin{aligned} f(y^{t+1}) - f(x^t) \leq \left(\frac{1 + \delta_{|F_t|+|L_t|}}{2} - \frac{1}{\eta}\right)\|y^{t+1}_{F_t}\|^2 \\ + \frac{1 + \delta_{|F_t|+|L_t|}}{2}\|x^t_{L_t}\|^2. \end{aligned} \tag{6}$$

Hence, the goal to show descent after PHT step, we need to show that for small enough η, $\|y^{t+1}_{F_t}\|^2$ is "large" and $\|x^t_{L_t}\|^2$ is "small". Our proof for each of the algorithm guarantees the same. Moreover, for Two-Stage(ℓ), we have an extra hard thresholding step as well, for which we need to show that increase in the overall objective function is not large.

Our specialized analysis for each of the algorithms follows the above mentioned proof outline, while guaranteeing that the RIP constant is not required to be small.

VI. Towards a General Analysis

We first provide a general set of lemmas that are then used appropriately to analyse each of the three algorithms: PHT(ℓ), Two-Stage(ℓ), and OMP. To this end, we first introduce some useful variables. Let $I^\star$ be the support set of $x^\star$. Define the sets

$$\begin{aligned} FA_t &= I_t \backslash I^\star && \text{(false alarms)} \\ MD_t &= I^\star \backslash I_t && \text{(missed detections)} \\ CO_t &= I_t \cap I^\star && \text{(correct detections)} \ . \end{aligned}$$

We first state two technical lemmas that we will need. These can be found in [14].

Lemma 8: For any $S \subset [n]$, we have,

$$\|I - A^T_S A_S\| \leq \delta_{|S|}.$$

Lemma 9: For any $S, T \subset [n]$ such that $S \cap T = \emptyset$, we have,

$$\|A^T_S A_T\|_2 \leq \delta_{|S \cup T|}.$$

We will also need the following result. A similar inequality was proved by Foucart [23] but we provide the full proof here for completeness.

Lemma 10: Let $b = Ax^\star$, where $I^\star = \operatorname{supp}(x^\star)$. Also, let $x = \operatorname{argmin}_{\operatorname{supp}(x)=I} \|Ax - b\|^2$. Then,

$$\begin{aligned} \sqrt{\|(x - x^\star)_{I \cap I^\star}\|^2 + \|x_{I \backslash I^\star}\|^2} &= \|(x - x^\star)_I\| \\ &\leq \frac{\delta_{|I \cup I^\star|}}{\sqrt{1 - \delta^2_{|I \cup I^\star|}}}\|x^\star_{I^\star \backslash I}\|. \end{aligned}$$

Proof: A similar inequality occurs in [23] and we rewrite the proof here. Since x_I is the solution to $\min_u \|A_I u - b\|^2$,

$$A_I^T(A_I x_I - b) = \mathbf{0}. \tag{7}$$

In the exact case, $b = Ax^\star$. Hence,

$$\|(x - x^\star)_I\|^2 = [(x - x^\star)_I \ \mathbf{0}] \begin{bmatrix} (x - x^\star)_I \\ -x^\star_{I^\star \setminus I} \end{bmatrix}. \tag{8}$$

Now, using (7):

$$0 = [(x - x^\star)_I \ \mathbf{0}] A_G^T A_G \begin{bmatrix} (x - x^\star)_I \\ -x^\star_{I^\star \setminus I} \end{bmatrix}, \tag{9}$$

where $G = [I \ I^\star \setminus I]$. Subtracting (9) from (8) we get,

$$\begin{aligned} \|(x - x^\star)_I\|^2 &= [(x - x^\star)_I \ \mathbf{0}](I - A_G^T A_G) \begin{bmatrix} (x - x^\star)_I \\ -x^\star_{I^\star \setminus I} \end{bmatrix} \\ &\leq \delta_{2k} \|(x - x^\star)_I\| \sqrt{\|(x - x^\star)_I\|^2 + \|x_{I^\star \setminus I}\|^2}, \end{aligned} \tag{10}$$

where the second inequality follows using Lemma 8. Lemma follows by just rearranging terms now. ∎

We now derive a fundamental lemma that shows that $z^{t+1}_{MD_t}$—elements of gradient descent iterate (z_{t+1}) over missed detections from previous step I_t—has large norm.

Lemma 11: Let $b = Ax^* + e$, where $e \in \mathbb{R}^m$ is the "noise" vector. Let $f(x^t) \geq \frac{C}{2}\|e\|^2$ and $\delta_{2k} < 1 - \frac{1}{2D\gamma}$, where $D = \frac{C - \sqrt{C}}{(\sqrt{C}+1)^2}$. Let $\gamma > 0$ be any constant. Then,

$$\gamma^2 \|A^T_{MD_t} A(x^t - x^*)\|^2 - \|x^t_{FA_t}\|^2 \geq c f(x^t),$$

where $c = 2\frac{(\sqrt{C}+1)^2}{C}(2\gamma D - \frac{1}{1-\delta_{2k}}) > 0$.

In particular, if $e = 0$ (i.e., noiseless case), we have:

$$0 < (4\gamma - \frac{2}{1 - \delta_{2k}}) f(x^t) \leq \gamma^2 \|A^T_{MD_t} A(x^t - x^*)\|^2 - \|x^t_{FA_t}\|^2,$$

where $\delta_{2k} < 1 - \frac{1}{2\gamma}$.

Proof: Since $x^t_{I_t}$ is the solution to the least squares problem $\min_x \|A_{I_t} x_{I_t} - b\|^2$,

$$A^T_{I_t}(A_{I_t} x^t_{I_t} - b) = 0. \tag{11}$$

Now, note that

$$\begin{aligned} f(x^t) &= \frac{1}{2}\|A_{I_t} x^t_{I_t} - b\|^2 \\ &= \frac{1}{2}((x^t_{I_t})^T A^T_{I_t}(A_{I_t} x^t_{I_t} - b) - b^T(A_{I_t} x^t_{I_t} - b)) \\ &= -\frac{1}{2} b^T (A_{I_t} x^t_{I_t} - b) \\ &= -\frac{1}{2}(x^*_{MD_t})^T A^T_{MD_t}(A_{I_t} x^t_{I_t} - b) - \frac{1}{2} e^T (A_{I_t} x^t_{I_t} - b), \end{aligned} \tag{12}$$

where the third equality follows from (11).

Now,

$$\begin{aligned} &\|x^\star_{MD_t} + \gamma A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\|^2 \\ &= \|x^\star_{MD_t}\|^2 + \gamma^2 \|A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\|^2 \\ &\quad + 2\gamma (A^T_{MD_t}(A_{I_t} x^t_{I_t} - b))^T x^\star_{MD_t} \\ &= \|x^\star_{MD_t}\|^2 + \gamma^2 \|A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\|^2 - 4\gamma f(x^t) \\ &\quad + 2\gamma e^T (A_{I_t} x^t_{I_t} - b). \end{aligned} \tag{13}$$

Using the above equality and using $\|x^\star_{MD_t} + \gamma A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\| \geq 0$, we have:

$$\begin{aligned} 0 &\leq \|x^\star_{MD_t}\|^2 + \|x^t_{FA_t}\|^2 - \|x^t_{FA_t}\|^2 \\ &\quad + \gamma^2 \|A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\|^2 - 4\gamma \left(f(x^t) + \frac{1}{2} e^T (A_{I_t} x^t_{I_t} - b) \right) \\ &\leq \|x^\star_{MD_t}\|^2 + \|x^t_{FA_t}\|^2 + \|x^t_{CO_t} - x^\star_{CO_t}\|^2 - \|x^t_{FA_t}\|^2 \\ &\quad + \gamma^2 \|A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\|^2 - 4\gamma \left(f(x^t) + \frac{1}{2} e^T (A_{I_t} x^t_{I_t} - b) \right) \\ &\overset{\zeta_1}{=} \|x^t - x^*\|^2 - \|x^t_{FA_t}\|^2 + \gamma^2 \|A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\|^2 \\ &\quad - 4\gamma \left(f(x^t) + \frac{1}{2} e^T (A_{I_t} x^t_{I_t} - b) \right) \\ &\overset{\zeta_2}{\leq} \frac{1}{1 - \delta_{2k}} \|A(x^t - x^*)\|^2 - \|x^t_{FA_t}\|^2 \\ &\quad + \gamma^2 \|A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\|^2 - 4\gamma \left(f(x^t) + \frac{1}{2} e^T (A_{I_t} x^t_{I_t} - b) \right) \\ &= \frac{1}{1 - \delta_{2k}} \|A(x^t - x^*)\|^2 - \|x^t_{FA_t}\|^2 \\ &\quad + \gamma^2 \|A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\|^2 - 4\gamma (1 - \frac{1}{\sqrt{C}}) f(x^t), \end{aligned} \tag{14}$$

where ζ_1 follows from $\|x^t - x^*\|^2 = \|x^\star_{MD_t}\|^2 + \|x^t_{FA_t}\|^2 + \|x^t_{CO_t} - x^\star_{CO_t}\|^2$, ζ_2 follows from RIP. The last inequality follows from:

$$|e^T(A_{I_t} x^t_{I_t} - b)| \leq \|e\| \|A_{I_t} x^t_{I_t} - b\| = \|e\| \cdot \sqrt{2 f(x^t)} \leq \frac{2}{\sqrt{C}} f(x^t).$$

The last inequality above follows from the assumption that $f(x^t) \geq \frac{C}{2}\|e\|^2$.

Similarly, using $f(x^t) \geq \frac{C}{2}\|e\|^2$, we have:

$$\begin{aligned} \|A(x^t - x^*)\| &\leq \|A(x^t - x^*) - e\| + \|e\|, \\ \|A(x^t - x^*)\|^2 &\leq 2(1 + \frac{1}{\sqrt{C}})^2 f(x^t). \end{aligned} \tag{15}$$

Using (14) and (15), we have:

$$\begin{aligned} 2\left(2\gamma(1 - \frac{1}{\sqrt{C}}) - \frac{1}{1 - \delta_{2k}}(1 + \frac{1}{\sqrt{C}})^2 \right) f(x^t) \\ \leq \gamma^2 \|A^T_{MD_t}(A_{I_t} x^t_{I_t} - b)\|^2 - \|x^t_{FA_t}\|^2. \end{aligned}$$

Now, by assumption $\delta_{2k} < 1 - \frac{1}{2D\gamma}$, where $D = \frac{(\sqrt{C}+1)^2}{C - \sqrt{C}}$. Hence, $c = 2\frac{(\sqrt{C}+1)^2}{C}(2\gamma D - \frac{1}{1-\delta_{2k}}) > 0$. ∎

VII. General Analysis

The results in the previous section allow us to quantify progress made in a (partial) hard thresholding step. This, in turn, permits a unified analysis of all iterative algorithms discussed in the paper.

Lemma 12 (Hard Thresholding): Suppose w^{t+1} is a k'-sparse iterate obtained as a result of solving a least squares problem on $J_{t+1} = \text{supp}(w^{t+1})$. Define

$$v^{t+1} = H_k(w^{t+1})$$

where $k < k'$. Then, we have,

$$f(v^{t+1}) - f(w^{t+1}) \leq \frac{(1 + \delta_\ell)\delta^2_{2k+\ell}}{1 - \delta_{2k+\ell}} f(w^{t+1})$$

where $\ell = k' - k$.

Proof: Let $I_{t+1} = \text{supp}(v^{t+1})$. As H_k is the hard thresholding operator, we have:

$$\|v^{t+1} - w^{t+1}\|^2 \leq \|x^\star - w^{t+1}\|^2 \leq \frac{2}{1-\delta_{2k+\ell}} f(w^{t+1}).$$

Moreover,

$$\begin{aligned} f(v^{t+1}) - f(w^{t+1}) &= (v^{t+1} - w^{t+1})^T A^T (Aw^{t+1} - b) \\ &\quad + \tfrac{1}{2}\|Av^{t+1} - Aw^{t+1}\|^2 \\ &= \frac{1}{2}\|Av^{t+1} - Aw^{t+1}\|^2 \\ &\leq \frac{1+\delta_\ell}{2}\|v^{t+1} - w^{t+1}\|^2, \end{aligned} \tag{16}$$

where the second equation follows as w^{t+1} is a least squares solution, and supports of both v^{t+1} and w^{t+1} are subsets of J_{t+1}. The last inequality follows from RIP.

Furthermore, $|J_{t+1} \backslash I_{t+1}| = \ell \leq |J_{t+1} \backslash I^\star|$. Hence, by definition of I_{t+1},

$$\|w^{t+1}_{J_{t+1}\backslash I_{t+1}}\|^2 \leq \|w^{t+1}_{J_{t+1}\backslash I^\star}\|^2.$$

Using above equation and Lemma 10, we have:

$$\|w^{t+1}_{J_{t+1}\backslash I_{t+1}}\|^2 \leq \frac{2\delta^2_{2k+\ell}}{1-\delta_{2k+\ell}} f(w^{t+1}). \tag{17}$$

Lemma now follows by using (16) and (17). ∎

In the lemma below, note that the sets F_t (found set), L_t (lost set), and MD_t (set of missed detections) are as they were defined in Section V and Section VI above.

Lemma 13 (Partial Hard Thresholding): Suppose x^t is a k-sparse iterate obtained as a result of solving a least squares problem on $I_t = \text{supp}(x^t)$. Define the gradient descent iterate

$$z^{t+1} = x^t - \eta A^T(Ax^t - b)$$

and let

$$y^{t+1} = PHT_{k'}\left(z^{t+1}; I_t, \ell\right)$$

where $k' \geq k, \ell \leq n - k$. Then, we have,

$$\begin{aligned} f(t^{t+1}) - f(x^t) \leq &\left(\frac{1+\delta_{|F_t|+|L_t|}}{2} - \frac{1}{\eta}\right)\|y_{F_t}\|^2 \\ &+ \frac{1+\delta_{|F_t|+|L_t|}}{2}\|x^t_{L_t}\|^2. \end{aligned}$$

Assume that $k' \geq k$. Also, assume that if $\ell > 0$, then $\delta_{2k} < 1 - 1/2\eta$. Then, we have:

$$\begin{aligned} &f(y^{t+1}) - f(x^t) \\ &\leq \left((1+\delta_{|F_t|+|L_t|})(\frac{1}{2}\mathbb{I}[\ell = 0] + \mathbb{I}[\ell \neq 0]) - \frac{1}{\eta}\right) \times \\ &\quad 2\eta^2(1-\delta_{2k})\min(1, \frac{\ell + k' - k}{|MD_t|}) f(x^t). \end{aligned}$$

Note that $|MD_t| \leq k$.

Proof: Since f is quadratic, second-order Taylor expansion is exact. Hence, we have,

$$\begin{aligned} f(y^{t+1}) - f(x^t) &= (y^{t+1} - x^t)^T A^T A(x^t - x^\star) \\ &\quad + \tfrac{1}{2}\|A(y^{t+1} - x^t)\|^2 \\ &\leq (y^{t+1} - x^t)^T A^T A(x^t - x^\star) \\ &\quad + \frac{1+\delta_{|F_t|+|L_t|}}{2}(\|y^{t+1}_{F_t}\|^2 + \|x^t_{L_t}\|^2). \end{aligned} \tag{18}$$

where the second inequality follows by using the fact that $y^{t+1}_{R_t} = x^t_{R_t}$ and using RIP of order $|F_t| + |L_t|$.

Since $x^t_{I_t}$ is obtained using least squares,

$$A^T_{I_t} A(x^t - x^\star) = \mathbf{0}.$$

Thus, $A^T_{L_t} A(x^t - x^\star) = \mathbf{0}$, because $L_t \subseteq I_t$. Next, note that

$$y^{t+1}_{F_t} = -\eta A^T_{F_t} A(x^t - x^\star).$$

Hence,

$$\begin{aligned} f(y^{t+1}) - f(x^t) \leq &\left(\frac{1+\delta_{|F_t|+|L_t|}}{2} - \frac{1}{\eta}\right)\|y^{t+1}_{F_t}\|^2 \\ &+ \frac{1+\delta_{|F_t|+|L_t|}}{2}\|x^t_{L_t}\|^2. \end{aligned} \tag{19}$$

Now, $|L_t| \leq \ell$. Hence, if $\ell = 0$ then, $|L_t| = 0$. This observation with (19) proves the first part of the lemma. We now consider three exhaustive cases:

1) $|F_t| < \ell + k' - k$ and $|F_t| < |MD_t|$: Note that if $\ell = 0$, then $|F_t| = k' - k$. Hence, this case does not apply.
Now assume $\ell > 0$. Also, $|F_t| - |L_t| = k' - k$. Hence, $|L_t| < \ell$. Assuming $\delta_{2k} < 1 - 1/2\eta, f(x^t) > 0$ and using Lemma 11,

$$\eta^2\|A^T_{MD_t} r_t\|^2 \geq \|x^t_{FA_t}\|^2 + 2(2\eta - \frac{1}{1-\delta_{2k}}) f(x^t).$$

Hence, if $\delta_{2k} < 1 - 1/2\eta$, at least one element of MD_t will be selected in F_t and similarly at least one element of FA_t will be selected in L_t.
Let $S \subseteq MD_t \backslash F_t$, s.t., $|S| = |F_t| - |MD_t \cap F_t|$. Now,

$$|S \cup (MD_t \cap F_t)| = |F_t|, |(MD_t \backslash F_t) \backslash S| = |MD_t| - |F_t|.$$

As y_{F_t} consists of top $|F_t|$ elements of $z^{t+1}_{MD_t}$, we have:

$$\|z^{t+1}_{S\cup(MD_t\cap F_t)}\|^2 \leq \|y_{F_t}\|^2. \tag{20}$$

Furthermore, since $|F_t| < \ell + k' - k$, hence every element of $z^{t+1}_{MD_t \backslash F_t}$ is smaller in magnitude than every element of $x^t_{FA_t \backslash L_t}$, otherwise that element should have been included in F_t. Furthermore, $|MD_t| - |F_t| \leq |FA_t| - |L_t| \leq |FA_t \backslash L_t|$. Hence,

$$\|z^{t+1}_{(MD_t\backslash F_t)\backslash S}\|^2 \leq \|x^t_{FA_t\backslash L_t}\|^2 \leq \|x^t_{FA_t}\|^2. \tag{21}$$

Adding (20) and (21), we get:

$$\|z^{t+1}_{MD_t}\|^2 \leq \|y^{t+1}_{F_t}\|^2 + \|x^t_{FA_t}\|^2. \tag{22}$$

Using the above equation along with Lemma 11 with $\gamma = \frac{1}{1-\delta_{2k}}$, we get:

$$\|y^{t+1}_{F_t}\|^2 \geq 2\eta^2(1-\delta_{2k}) f(x^t). \tag{23}$$

2) $|F_t| = \ell + k' - k < |MD_t|$: By definition of $y_{F_t}^{t+1}$:

$$\frac{\|y_{F_t}^{t+1}\|^2}{|F_t|} \geq \frac{\|z_{MD_t}^{t+1}\|^2}{|MD_t|}.$$

Hence, using Lemma 11 with $\eta = \gamma$ and the fact that $|F_t| = \ell + k' - k$:

$$\|y_{F_t}^{t+1}\|^2 \geq \frac{\ell + k' - k}{k} 2\eta^2(1 - \delta_{2k}) f(x^t), \tag{24}$$

as $|MD_t| \leq k$.

3) $|F_t| \geq |MD_t|$: Since, $y_{F_t}^{t+1}$ is the top most elements of z^{t+1}. Hence, assuming $|F_t| \geq |MD_t|$,

$$\|y_{F_t}^{t+1}\|^2 \geq \|z_{MD_t}^{t+1}\|^2.$$

Now, using Lemma 11 with $\gamma = 1/(1 - \delta_{2k})$, we have:

$$\|y_{F_t}^{t+1}\|^2 \geq 2\eta^2(1 - \delta_{2k}) f(x^t). \tag{25}$$

Note that out of the above three cases, the condition $\delta_{2k} < 1 - 1/(2\eta)$ is required only for the first case, where $\ell > 0$ necessarily.

We now get the lemma by combining bounds for all the three cases, i.e., (23), (24), (25) along with the above mentioned observation. ∎

A. Proof of PHT(ℓ) Family

Proof of Theorem 2: Recall that for PHT(ℓ) family, the Partial Hard Thresholding operator is applied with $k' = k$ and $0 < \ell \leq k$. That is, $|F_t| = |L_t| \leq \ell$. Hence, using Lemma 13, we have:

$$f(y^{t+1}) - f(x^t) \leq \left(1 + \delta_{2\ell} - \frac{1}{\eta}\right) \eta^2(1 - \delta_{2k}) \frac{\ell}{k} f(x^t),$$

assuming $\delta_{2k} < 1 - 1/2\eta$. Theorem now follows by observing that $f(x^{t+1}) \leq f(y^{t+1})$ and by setting $\eta = \frac{c}{1+\delta_{2\ell}}$, where $0 < c < 1$ is a constant that is close to 1. ∎

B. Proof of Two-stage(ℓ) Family

Proof of Theorem 4: Recall that $y^{t+1} = PHT_{k+\ell}\left(z^{t+1}; I_t, 0\right)$. Hence, using Lemma 13 with $\eta = \frac{1}{1+\delta_\ell}$, we have:

$$f(y^{t+1}) - f(x^t) \leq -\frac{(1 - \delta_{2k})}{(1 + \delta_\ell)} f(x^t).$$

Using the fact that $f(w^{t+1}) \leq f(y^{t+1})$, we have:

$$f(w^{t+1}) - f(x^t) \leq -\frac{(1 - \delta_{2k})}{(1 + \delta_\ell)} f(x^t). \tag{26}$$

Also, recall that $v^{t+1} = H_k\left(w^{t+1}\right)$. Hence, using Lemma 12, we have:

$$f(v^{t+1}) - f(w^{t+1}) \leq -\frac{(1 + \delta_\ell)\delta_{2k+\ell}^2}{1 - \delta_{2k+\ell}} f(w^{t+1}). \tag{27}$$

Hence, using (26), (27), and the fact that $f(x^{t+1}) \leq f(v^{t+1})$, we have:

$$\begin{aligned} f(x^{t+1}) &\leq \left(1 + \frac{(1 + \delta_\ell)\delta_{2k+\ell}^2}{1 - \delta_{2k+\ell}}\right) f(w^{t+1}) \\ &\leq \left(1 + \frac{(1 + \delta_\ell)\delta_{2k+\ell}^2}{1 - \delta_{2k+\ell}}\right) \frac{2\delta_{2k+\ell}}{1 + \delta_{2k+\ell}} f(x^t). \end{aligned}$$

That is, if $\delta_{2k+\ell} < .46$, then,

$$f(x^{t+1}) \leq 0.991 f(x^t).$$

∎

C. Proof of OMP

We now present a proof of OMP that holds under significantly weaker RIP conditions than that of [12] which requires $\delta_{31k} \leq 1/3$.

Proof of Theorem 7: Recall that $y^{t+1} = PHT_{k+t+1}\left(z^{t+1}; I_t, 0\right)$. Hence, using Lemma 13, we have:

$$f(y^{t+1}) - f(x^t) \leq -(1 - \delta_{2k+t+1}) \frac{f(x^t)}{|MD_t|}.$$

Note that, $(1 - \delta_{2k+t+1}) \frac{f(x^t)}{|MD_t|} \geq 0.5 \cdot (1 - \delta_{2k+1+1})^2$. Hence,

$$f(x^T) \leq f(x^0) - \frac{T}{2}(1 - \delta_{2k+T})^2 \leq \frac{1}{2}(1 + \delta_{2k+T})k - \frac{T}{2}(1 - \delta_{2k+T})^2.$$

That is, we need:

$$T(1 - \delta_{2k+T})^2 \geq (1 + \delta_{2k+T})k.$$

Setting $T = 2k$, we see that $\delta_{4k} \leq 0.2$ satisfies the above requirement. Similarly, setting $T = 3k$, we relax the requirement to be $\delta_{5k} \leq 0.33$. ∎

ACKNOWLEDGMENT

Part of AT's contribution to this research occurred while he was a postdoctoral fellow at UT Austin. ISD acknowledges support from the Moncrief Grand Challenge Award. This research was also supported by NSF grants CCF-1564000 (for ISD) and DMS-1612549 (for AT).

REFERENCES

[1] A. Maleki and D. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 330–341, 2010.

[2] S. Foucart, "Hard thresholding pursuit: an algorithm for compressive sensing," *SIAM Journal on Numerical Analysis*, vol. 49, no. 6, pp. 2543–2563, 2011.

[3] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[4] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[5] E. J. Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9-10, pp. 589–592, 2008.

[6] Q. Mo and S. Li, "New bounds on the restricted isometry constant δ_{2k}," *Applied and Computational Harmonic Analysis*, vol. 31, no. 3, pp. 460–468, 2011.

[7] D. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences USA*, vol. 106, no. 45, pp. 18 914–18 919, 2009.

[8] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *27th Annual Asilomar Conference on Signals, Systems, and Computers*, vol. 1, 1993, pp. 40–44.

[9] G. Davis, S. Mallat, and M. Avellaneda, "Greedy adaptive approximation," *Constructive Approximation*, vol. 13, pp. 57–98, 1997.

[10] H. Rauhut, "On the impossibility of uniform sparse reconstruction using greedy methods," *Sampling Theory in Signal and Image Processing*, vol. 7, no. 2, pp. 197–215, 2008.

[11] Q. Mo and Y. Shen, "Remarks on the restricted isometry property in orthogonal matching pursuit algorithm," 2011, preprint arXiv:1101.4458.
[12] T. Zhang, "Sparse recovery with orthogonal matching pursuit under RIP," *IEEE Transactions on Information Theory*, vol. 57, no. 9, pp. 6215–6221, 2011.
[13] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.
[14] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.
[15] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.
[16] A. Maleki, "Convergence analysis of iterative thresholding algorithms," in *Allerton Conference on Communication, Control and Computing*, 2009.
[17] T. Zhang, "Adaptive forward-backward greedy algorithm for sparse learning with linear models," in *Advances in Neural Information Processing Systems*, 2008.
[18] A. Demiriz, K. P. Bennett, and J. Shawe-Taylor, "Linear programming boosting via column generation," *Machine Learning*, vol. 46, no. 1-3, pp. 225–254, 2002.
[19] M. Rudelson and R. Vershynin, "On sparse reconstruction from fourier and gaussian measurements," *Communications on Pure and Applied Mathematics*, vol. 61, no. 8, pp. 1025–1045, 2008.
[20] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The convex geometry of linear inverse problems," *Foundations of Computational mathematics*, vol. 12, no. 6, pp. 805–849, 2012.
[21] D. Amelunxen, M. Lotz, M. B. McCoy, and J. A. Tropp, "Living on the edge: Phase transitions in convex programs with random data," *Information and Inference*, p. iau005, 2014.
[22] P. Jain, A. Tewari, and I. S. Dhillon, "Orthogonal matching pursuit with replacement," in *Advances in Neural Information Processing Systems 24*, 2011, pp. 1215–1223.
[23] S. Foucart, "Hard thresholding pursuit: an algorithm for compressive sensing," *SIAM Journal on Numerical Analysis*, vol. 49, no. 6, pp. 2543–2563, 2011.

PLACE PHOTO HERE

Prateek Jain is a member of the Machine Learning and Optimization and the Algorithms and Data Sciences Group at Microsoft Research, Bangalore, India. He is also an adjunct faculty member of the Department of Computer Science and Engineering at the Indian Institute of Technology, Kanpur, India. His research interests are in machine learning, large-scale (non-convex optimization), and statistical learning theory. He is also interested in applications of machine learning to privacy, computer vision, text mining and natural language processing. He completed his Ph.D. at the University of Texas at Austin under Prof. Inderjit S. Dhillon.

PLACE PHOTO HERE

Ambuj Tewari is an assistant professor in the Department of Statistics and the Department of EECS (by courtesy) at the University of Michigan, Ann Arbor. His is also affiliated with the Michigan Institute for Data Science (MIDAS). He obtained his Ph.D. under the supervision of Peter Bartlett at the University of California at Berkeley. His research interests lie in machine learning including statistical learning theory, online learning, reinforcement learning and control theory, network analysis, and optimization for machine learning. He collaborates with scientists to seek novel applications of machine learning in mobile health, learning analytics, and computational chemistry. His research has been recognized with paper awards at the COLT and AISTATS conferences. He has received an NSF CAREER award and a Sloan Research Fellowship.

PLACE PHOTO HERE

Inderjit S. Dhillon is the Gottesman Family Centennial Professor of Computer Science and Mathematics at University of Texas at Austin, where he is also the Director of the ICES Center for Big Data Analytics. His main research interests are in big data, machine learning, network analysis, linear algebra and optimization. He received his B.Tech. degree from the Indian Institute of Technology, Bombay, and Ph.D. from the University of California at Berkeley. Inderjit has received several awards, including the ICES Distinguished Research Award, the SIAM Outstanding Paper Prize, the Moncrief Grand Challenge Award, the SIAM Linear Algebra Prize, the University Research Excellence Award, and the NSF Career Award. He has published over 160 journal and conference papers, and has served on the Editorial Board of the Journal of Machine Learning Research, the IEEE Transactions of Pattern Analysis and Machine Intelligence, Foundations and Trends in Machine Learning and the SIAM Journal for Matrix Analysis and Applications. Inderjit is an ACM Fellow, an IEEE Fellow, a SIAM Fellow and an AAAS Fellow.