

# A Generalized Class of Hard Thresholding Algorithms for Sparse Signal Recovery

Jean-Luc Bouchot

**Abstract** We introduce a whole family of hard thresholding algorithms for the recovery of sparse signals  $\mathbf{x} \in \mathbb{C}^N$  from a limited number of linear measurements  $\mathbf{y} = \mathbf{A}\mathbf{x} \in \mathbb{C}^m$ , with  $m \ll N$ . Our results generalize previous ones on hard thresholding pursuit algorithms. We show that uniform recovery of all  $s$ -sparse vectors  $\mathbf{x}$  can be achieved under a certain restricted isometry condition. While these conditions might be unrealistic in some cases, it is shown that with high probability, our algorithms select a correct set of indices at each iteration, as long as the active support is smaller than the actual support of the vector to be recovered, with a proviso on the shape of the vector. Our theoretical findings are illustrated by numerical examples.

## 1 Compressive Sensing and Sparse Signal Recovery

This paper is concerned with the standard compressive sensing problem, i.e., we analyze the reconstruction of sparse signals  $\mathbf{x} \in \mathbb{C}^N$  based only on a few number of (linear) measurements  $\mathbf{y} \in \mathbb{C}^m$  where  $m \ll N$ . It is known from the compressive sensing literature that recovery of  $s$ -sparse signals  $\mathbf{x}$  is ensured when the sensing (or measurement) matrix is random (Gaussian or sub-Gaussian for instance) and when the number of measurements scales linearly with the sparsity of the signal up to a log factor. Known methods arise mainly from optimization theory such as the  $\ell_1$  minimization [YZ11, CR05], reweighted norm minimizations [CWB08, WN10], primal-dual optimization [CP11], or from iterative solvers (see for instance [Tro04, TG07, BD09, NT09]).

We investigate in particular some variations of the Hard Thresholding Pursuit (HTP) algorithm [Fou11], an iterative thresholding based method, and its graded approach, a recent variation that does not require prior knowledge of the spar-

---

Jean-Luc Bouchot,  
Department of Mathematics, Drexel University, 3141 Chestnut Street, Philadelphia, e-mail: jean-luc.bouchot@drexel.edu.

sity [BFH13]. We analyze the reconstruction abilities of these algorithms in both idealized and realistic settings. In particular we introduce a generalization that improve the speed performance of (GHTP).

The idealized setting is characterized by the fact that the signal to be recovered  $\mathbf{x}$  is exactly  $s$ -sparse and that the measurements occur in an error-free manner. In this case exact recovery is ensured by all such algorithms provided that a certain *restricted isometry condition (RIC)* is met by the sensing matrix [CT05, Fou12]. In comparison, we may consider a more realistic setting in which the vector  $\mathbf{x}$  may suffer a sparsity defect and the measurements through the matrix  $\mathbf{A}$  may be inaccurate. In this case, we have  $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$  where  $\mathbf{e} \in \mathbb{C}^m$  represents the error induced by the measurement process. The sparsity defect can be integrated into this error term by considering  $\mathbf{x} = \mathbf{x}_S + \mathbf{x}_{\bar{S}}$  where  $\mathbf{x}_S$  corresponds to the  $s$  most important components (i.e. the largest absolute entries) of  $\mathbf{x}$ . Thus we may incorporate the remaining components into the noise as  $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} = \mathbf{A}\mathbf{x}_S + (\mathbf{A}\mathbf{x}_{\bar{S}} + \mathbf{e}) = \mathbf{A}\mathbf{x}_S + \mathbf{e}'$  where  $\mathbf{e}' = \mathbf{A}\mathbf{x}_{\bar{S}} + \mathbf{e} \in \mathbb{C}^m$  contains both the sparsity defect and the measurement noise.

The remainder of this article is organized as follows. We start in section 2 by reviewing some previous work regarding the (HTP) and (GHTP) algorithms (subsection 2.1). This leads to introduce a family of algorithms that generalizes the two previous ones in subsection 2.2. These algorithms are studied theoretically in the following sections in both uniform (see Section 3) and nonuniform settings (Section 4). Finally, section 5 compares and validates numerically our findings.

Throughout this paper we use the following notations:

- $\mathbf{x}^*$  represents the nonincreasing rearrangement of a vector  $\mathbf{x}$ :

$$x_1^* \geq x_2^* \geq \dots \geq x_N^* \geq 0$$

and there exists a permutation  $\pi$  of  $\{1, \dots, N\}$  such that  $x_j^* = |x_{\pi(j)}|$ .

- $S$  is the support of an  $s$ -sparse vector  $\mathbf{x}$  or the set of indices of its  $s$  largest absolute entries.
- $\mathbf{x}_T$  corresponds to the vector  $\mathbf{x}$  either restricted to the set  $T$  or such that  $x_{Ti} = x_i$  for  $i \in T$  and 0 elsewhere, depending on the context.
- The complementary of a set  $T$  in  $\{1, \dots, N\}$  is denoted by  $\bar{T}$
- $\lceil \cdot \rceil$  and  $\lfloor \cdot \rfloor$  denote respectively the ceil and floor functions.
- $\delta_s$  corresponds to the restricted isometry constant of order  $s$  of a given matrix  $\mathbf{A}$  and is defined as the smallest  $\delta$  such that

$$(1 - \delta) \|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1 + \delta) \|\mathbf{x}\|_2^2$$

holds for any  $s$ -sparse vector  $\mathbf{x}$ .

## 2 (HTP), (GHTP) and their Generalizations

### 2.1 Previous Results

The Hard Thresholding Pursuit [Fou11] and its graded variant [BFH13] can be summarized by the following steps:

$$S^n := \{ \text{indices of } k \text{ largest entries of } |\mathbf{x}^{n-1} + \mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^{n-1})| \}, \quad (\text{GHTP}_1)$$

$$\mathbf{x}^n := \operatorname{argmin} \{ \|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2, \operatorname{supp}(\mathbf{z}) \subset S^n \}, \quad (\text{GHTP}_2)$$

with  $k = s$  for (HTP) and  $k = n$  for (GHTP).

It was shown that robust and stable recovery is achieved under some RIC:

**Theorem 1** *If the restricted isometry constant of the matrix  $\mathbf{A} \in \mathbb{C}^{m \times N}$  obeys*

$$\delta_{3s} \leq \frac{1}{3} \text{ for (HTP), and } \delta_{9s} \leq \frac{1}{3} \text{ for (GHTP),}$$

*then the sequences  $(\mathbf{x}^n)$  produced by (HTP) or (GHTP) with  $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} \in \mathbb{C}^m$  for some  $s$ -sparse  $\mathbf{x} \in \mathbb{C}^N$  and some  $\mathbf{e} \in \mathbb{C}^m$  with  $\|\mathbf{e}\|_2 \leq \gamma x_s^*$  satisfy*

$$\|\mathbf{x} - \mathbf{x}^{\bar{n}}\|_2 \leq d \|\mathbf{e}\|_2, \quad \bar{n} \leq cs.$$

*The constants  $c \leq 3$  for (HTP) and  $c \leq 4$  for (GHTP),  $d \leq 2.45$ , and  $\gamma \geq 0.079$  depend only on  $\delta_{3s}$  or  $\delta_{9s}$ .*

It is worth mentioning that reshuffling the index set in the (GHTP) algorithm adds robustness to (GHTP) (as seen in the numerical experiments in section 5) over *Orthogonal Matching Pursuit (OMP)* at the cost that its implementation cannot be done using QR updates.

### 2.2 Generalizations

We investigate here some generalizations of the Graded Hard Thresholding Pursuit that improve the speed of the algorithm while only slightly deteriorating its reconstruction capability. In order to speed up the convergence we need to lower the number of iterations. Following an index selection process similar to the (HTP) and (GHTP) algorithms, we introduce ( $f$ -HTP) that relies on a different number of indices selected per iteration (note that Generalized HTP would be a confusing name with regards to the (GHTP) algorithm).

Let  $f : \mathbb{N} \rightarrow \mathbb{N}$  be a non-decreasing function such that there exists  $n_0 \geq 0$  with  $f(n) \geq s$  for any  $n \geq n_0$ . The ( $f$ -HTP) algorithm is defined by the following sequence of operations:

$$S^n := \{\text{indices of } f(n) \text{ largest entries of } |\mathbf{x}^{n-1} + \mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^{n-1})|\}, \quad (f\text{-HTP}_1)$$

$$\mathbf{x}^n := \operatorname{argmin}\{\|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2, \operatorname{supp}(\mathbf{z}) \subset S^n\}. \quad (f\text{-HTP}_2)$$

Observe that the constant function  $f(n) = s$  yields the original (HTP) algorithm while  $f(n) = n$  corresponds to (GHTP). Particularly interesting in terms of speed and number of iterations is the case  $f(n) = 2^{n-1}$  which shall be refer to as (GHTP<sup>2</sup>).

### 2.3 First Results

We first provide some preliminary results as we did for the original graded algorithm (GHTP) in [BFH13]. We show that a similar geometric decay of the error at each iteration  $\|\mathbf{x} - \mathbf{x}^n\|_2$  holds for the generalization ( $f$ -HTP), see Equation (3). It also ensures that a certain number of indices of largest entries may be included in the support after a given number of iterations (see Lemma 1). These results will allow us to prove the main result for uniform recovery (as stated in Theorem 2) by induction.

#### 2.3.1 Geometric Decay

In the remainder of this article we will define  $n_0$  as the smallest integer such that  $f(n) \geq s$ , for all  $n \geq n_0$ . In particular,  $n_0 = 0$  for (HTP),  $s$  for (GHTP) and  $\lceil \log_2(s) \rceil$  for (GHTP<sup>2</sup>). Using the results from [Fou11, BFH13] we have the following estimates, for  $n \geq n_0$ :

$$\begin{aligned} \|\mathbf{x}^{n+1} - \mathbf{x}\|_2 &\leq \sqrt{\frac{1}{1 - \delta_{f(n+1)+s}^2}} \left\| (\mathbf{x}^{n+1} - \mathbf{x})_{S^{n+1}} \right\|_2 \\ &\quad + \frac{1}{1 - \delta_{f(n+1)+s}} \|(\mathbf{A}^* \mathbf{e})_{S^{n+1}}\|_2, \end{aligned} \quad (1)$$

$$\left\| (\mathbf{x}^{n+1} - \mathbf{x})_{S^{n+1}} \right\|_2 \leq \sqrt{2} \delta_{f(n)+f(n+1)+s} \|\mathbf{x}^n - \mathbf{x}\| + \sqrt{2} \|(\mathbf{A}^* \mathbf{e})_{S\Delta S^{n+1}}\|_2. \quad (2)$$

Combining these two estimates yields the geometric decay

$$\|\mathbf{x}^{n+1} - \mathbf{x}\|_2 \leq \sqrt{\frac{2\delta_{f(n)+f(n+1)+s}^2}{1 - \delta_{f(n+1)+s}^2}} \|\mathbf{x}^n - \mathbf{x}\|_2 + \tau_{f(n+1)+s} \|\mathbf{e}\|_2 \quad (3)$$

with  $\tau_{f(n+1)+s} = \sqrt{\frac{2}{(1 - \delta_{f(n+1)+s})^2}} + \frac{\sqrt{1 + \delta_{f(n+1)+s}}}{1 - \delta_{f(n+1)+s}}$ . These results are the same as in our previous paper up to the RIC that needs to be adapted. In a more concise way we can write, where the multiplicative coefficient can be written depending only on  $\delta_{2f(n+1)+s}$ :

$$\|\mathbf{x}^{n+1} - \mathbf{x}\|_2 \leq \rho_{2f(n+1)+s} \|\mathbf{x}^n - \mathbf{x}\|_2 + \tau_{f(n+1)+s} \|\mathbf{e}\|_2$$

$$\text{with } \rho_{2f(n+1)+s} = \sqrt{\frac{2\delta_{2f(n+1)+s}^2}{1 - \delta_{2f(n+1)+s}^2}}.$$

### 2.3.2 Preparatory Lemma

As for the original (GHTP) algorithm [BFH13] we can show that, if the  $p$  largest absolute entries are contained in the support at iteration  $n$ , then  $k$  further iterations of ( $f$ -HTP) are sufficient to recover the  $q$  following largest entries, as stated in the following lemma.

**Lemma 1** *Let  $\mathbf{x} \in \mathbb{C}^N$  be  $s$ -sparse and let  $(S^n)$  be the sequence of index sets produced by ( $f$ -HTP) with  $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$  for some  $\mathbf{e} \in \mathbb{C}^m$ . For integers  $p \geq 0$  and  $n \geq n_0$ , suppose that  $S^n$  contains the indices of  $p$  largest absolute entries of  $\mathbf{x}$ . Then, for integers  $k, q \geq 1$ ,  $S^{n+k}$  contains the indices of  $p+q$  largest absolute entries of  $\mathbf{x}$ , provided*

$$x_{p+q}^* > \rho_{s+2f(n+k)}^k \|\mathbf{x}_{\{p+1, \dots, s\}}^*\|_2 + \kappa_{n+k-1} \|\mathbf{e}\|_2, \quad (4)$$

with the constants  $\rho_{n+k-1}$  as defined above and  $\kappa_{n+k-1} = \frac{\sqrt{2}\delta_{s+f(n+k-1)}\sqrt{1-\delta_{s+f(n-1)}}}{1-\delta_{s+f(n-2)}} + \frac{\sqrt{2}}{1-\delta_{s+f(n+1)}} \frac{\delta_{s+f(n+k-1)}}{1-\rho_{n+k-1}} + \sqrt{2}\sqrt{1+\delta_2}$  depending only on the restricted isometry constant  $\delta_{s+f(n+k)}$ .

**Remark 1** *The proof of Lemma 1 is not provided here. It follows directly from the proof of Lemma 3 and 4 from [BFH13] with changes imposed on the current iteration number and the number of indices selected, which is replaced by  $f(n)$  everywhere.*

**Remark 2** *Lemma 1 is not ideal in the sense that the number of iterations needed for the recovery of the next  $q$  largest entries, does not depend on the actual index selection method and whether we select exponentially many new indices at each iteration or just a linear number of new candidates. This leads to overestimate the number of iterations needed. As we see in the following section it creates RIC that are not always realistic, as they yield RIP of order up to  $s^s$  (in the worst, but fastest, scenario). It shows however, that there exist conditions under which the convergence of the algorithm is guaranteed. We also see that in particular cases (namely when dealing with power or flat vectors) the conditions from Theorem 2 can be drastically improved. Moreover, as suggested by the numerical experiments in Section 5, these results are only a rough overestimation of the actual number of iterations needed.*

### 3 Uniform Recovery via ( $f$ -HTP)

#### 3.1 General Results

This section is dedicated to the problem of uniform recovery of all  $s$ -sparse vectors  $\mathbf{x}$  given a certain sensing matrix. While this gives some ideas of why the ( $f$ -HTP) algorithms may converge, our proof yields, for certain choices of  $f$ , unrealistic and unapplicable conditions. Such considerations are detailed in Table 1.

**Theorem 2** *If the restricted isometry constant of the matrix  $\mathbf{A} \in \mathbb{C}^{m \times N}$  obeys*

$$\delta_{2f(\bar{n})+n_0} \leq \frac{1}{3},$$

*then the sequence  $(\mathbf{x}^n)$  produced by ( $f$ -HTP) with  $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} \in \mathbb{C}^m$  for some  $s$ -sparse  $\mathbf{x} \in \mathbb{C}^N$  and some  $\mathbf{e} \in \mathbb{C}^m$  with  $\|\mathbf{e}\|_2 \leq \gamma x_s^*$  satisfies*

$$\|\mathbf{x} - \mathbf{x}^{\bar{n}}\|_2 \leq d\|\mathbf{e}\|_2, \quad \bar{n} \leq cs.$$

*The constants  $c \leq 4$ ,  $d \leq 2.45$ , and  $\gamma \geq 0.079$  depend only on  $\delta_{s+2f(\bar{n})}$ .*

**Table 1** Examples of RIC that a sensing matrix should fulfil for uniform recovery, according to Theorem 2

$f$	$s$	$n$	$2n$	$n(n+1)/2$	$2^{n-1}$
RIC	$3s^1$	$9s$	$s/2 + 16s$	$\sqrt{2s + 8s^2}$	$s + s^s$
Name	(HTP)	(GHTP)			(GHTP <sup>2</sup> )

This theorem generalizes the one obtained first for (HTP) and (GHTP) to more general index selection schemes. It is purely an adaptation of our previous results and does not depend on the index selection function  $f$ . Therefore, as stated above it generates unrealistic restricted isometry conditions. For instance, when considering the case of  $f(n) = 2^{n-1}$  we would need to ensure an RIC of order  $\Omega(s^s)$ . Table 1 gives some examples of RIC for different choices of  $f$ .

While the two last conditions are unrealistic, the first cases still yield reasonable RIC. For instance the case  $f = 2n$  yields an RIC at the order  $16s$  which is still in a comparable range as for (OMP) (see [Zha11] for a stable recovery under RIC of order  $13s$ ).

Fortunately these strong conditions can be improved in the particular cases of power vector or almost flat vectors. The recovery of power vectors is analyzed in the following section where we show in the case of (GHTP<sup>2</sup>) that the matrix only needs to obey an RIC of order  $\Omega(s^{\text{polylog}(s)})$ . (Other examples are given in Table 2.)

<sup>1</sup> This result actually coincides with the one from the original paper about (HTP) [Fou11]

### 3.2 The Case of Power Vectors

We investigate the convergence of the family of generalized algorithms when facing particular power vectors. Our results rely on the following lemma used for decomposing the support:

**Lemma 2** Any set  $S \subseteq \{1, \dots, N\}$  of size  $s \leq N$  can be decomposed in  $r$  subsets  $S_1, \dots, S_r$  such that

1.  $r = \lceil \log_2(s) \rceil + 1$
2.  $S = \bigcup_{i=1}^r S_i$
3.  $S_i \cap S_j = \emptyset$ , for  $j \neq i$
4.  $|S_i| \leq \lceil s/2^i \rceil$

*Proof.* We show this result by induction on the set size  $s$ . For  $s = 1$ ,  $S_1 = S$  fulfills all the criteria. Assume now that Lemma 2 holds for all  $1 \leq n \leq s - 1$ . Without loss of generality, we can consider the set  $S = \{1, \dots, s\}$ . It holds  $S = S_1 \cup T$  with  $S_1 = \{1, \dots, \lceil s/2 \rceil\}$  and  $T = S \setminus S_1$ . And we have  $|T| = s - \lceil s/2 \rceil < s$  and therefore, applying the induction hypothesis yields  $T = \bigcup_{j=1}^{r_T} T_j$  with  $r_T = \lceil \log_2(|T|) \rceil + 1$  and  $|T_j| \leq \lceil |T|/2^j \rceil$ . We now define  $S_i := T_{i-1}$  for  $i > 1$  and therefore the partition  $S_1, \dots, S_r$  fulfils the three first criteria of the lemma. To verify the last statement of the lemma we consider two separated cases:

**If  $s$  is even**, then there exists a  $k \in \mathbb{N}$  such that  $s = 2k$  and  $|S_1| = |T| = k$ . The induction hypothesis implies, for  $j \geq 1$

$$|T_j| \leq \lceil k/2^j \rceil \quad \text{and} \quad |S_{j+1}| \leq \lceil s/2^{j+1} \rceil$$

which proves the last point of the lemma.

**If  $s$  is odd**, then there exists a  $k \in \mathbb{N}$  such that  $s = 2k + 1$  and  $|S_1| = k + 1$  and  $|T| = k$ . The induction hypothesis implies, for  $j \geq 1$

$$|T_j| \leq \lceil k/2^j \rceil \quad \text{and} \quad |S_{j+1}| \leq \lceil s/2^{j+1} - 1/2^{j+1} \rceil \leq \lceil s/2^{j+1} \rceil$$

which finishes the proof of the lemma.  $\square$

Consider vectors  $\mathbf{x}$  such that for all  $1 \leq j \leq s$ ,  $x_j^* = 1/j^\alpha$  for some  $\alpha > 1/2$  (first, other cases will be considered later). We have

$$\begin{aligned} \|\mathbf{x}_{\{p+1, \dots, s\}}^*\|_2^2 &= \frac{1}{(p+1)^{2\alpha}} + \dots + \frac{1}{s^{2\alpha}} \\ &\leq \int_p^s \frac{1}{x^{2\alpha}} dx = \frac{1}{2\alpha-1} \left( \frac{1}{p^{2\alpha-1}} - \frac{1}{s^{2\alpha-1}} \right) \leq \frac{1}{2\alpha-1} \frac{1}{p^{2\alpha-1}}. \end{aligned}$$

With this, it is sufficient to find  $k$  and  $q$  such that

$$\frac{1}{(p+q)^{2\alpha}} > \rho^{2k} \frac{1}{2\alpha-1} \frac{1}{p^{2\alpha-1}},$$

for condition (4) from Lemma 1 to be valid.

This condition is equivalent to

$$k > \frac{1}{\log(1/\rho^2)} \log \left( \frac{p}{2\alpha - 1} \left( 1 + \frac{q}{p} \right)^{2\alpha} \right).$$

In conclusion,  $\{1, \dots, p\} \subset S^n \Rightarrow \{1, \dots, p+q\} \subset S^{n+k}$ . holds provided that

$$k > \frac{2\alpha \log \left( p \left( 1 + \frac{q}{p} \right) \right)}{\log(1/\rho^2)} - \frac{\log(2\alpha - 1)}{\log(1/\rho^2)}$$

If we now consider  $r$  subsets  $S_1, \dots, S_r$ ,  $r = \lfloor \log_2(s) \rfloor + 1$  as suggested by Lemma 2, then we can successively apply Lemma 1 to each  $r$  subsets  $S_i$ . Defining  $S_0 = \emptyset$ ,  $q_i = |S_i|$ , for  $i \geq 0$ ,  $k_i$  the number of iterations needed to add subset  $S_i$ , using  $k_0 = n_0$ , and  $p_i = \sum_{j=1}^{i-1} q_j$ , we finally get that the number of iterations for uniform recovery is bounded by

$$\begin{aligned} \bar{n} &\leq \sum_{i=0}^r k_i \leq \frac{2\alpha}{\log(1/\rho^2)} \sum_{i=1}^r \log \left( p_i \left( 1 + \frac{q_i}{p_i} \right) \right) - r \frac{\log(2\alpha - 1)}{\log(1/\rho^2)} + n_0 \\ &\leq \frac{2\alpha}{\log(1/\rho^2)} \sum_{i=1}^r \log \left( \sum_{j=1}^i q_j \right) - r \frac{\log(2\alpha - 1)}{\log(1/\rho^2)} + n_0 \\ &\leq \frac{2\alpha}{\log(1/\rho^2)} \sum_{i=1}^r \log \left( \sum_{j=1}^i (s/2^j + 1) \right) - r \frac{\log(2\alpha - 1)}{\log(1/\rho^2)} + n_0 \quad (5) \\ &\leq \frac{2\alpha}{\log(1/\rho^2)} \sum_{i=1}^r \log \left( s \sum_{j=1}^i 1/2^j + i \right) - r \frac{\log(2\alpha - 1)}{\log(1/\rho^2)} + n_0 \\ &\leq \frac{2\alpha}{\log(1/\rho^2)} \sum_{i=1}^r \log(2s) - r \frac{\log(2\alpha - 1)}{\log(1/\rho^2)} + n_0, \end{aligned}$$

where we have used the fact that  $q_j \leq \lceil s/2^j \rceil \leq s/2^j + 1$  in inequality (5). With such a partition we have that  $r = \lfloor \log_2(s) \rfloor + 1 \leq \log_2(s) + 1$  and hence  $\bar{n}$  can be bounded by

$$\bar{n} \leq (\log_2(s) + 1) \left( \log(2s) \frac{2\alpha}{\log(1/\rho^2)} - \frac{\log(2\alpha - 1)}{\log(1/\rho^2)} \right) + n_0.$$

Using this, we only need to ensure the RIC to the order  $2^{\bar{n}}$  which is  $\Omega(s \cdot s^{\log(s)})$  when using the (GHTP<sup>2</sup>). This is not yet acceptable for real-world applications but much less critical than what Theorem 2 suggests. Moreover, it corresponds also to the worst case scenario for ( $f$ -HTP) algorithms.

If we now consider **the case**  $\alpha = 1/2$ , a similar analysis yields

$$\|\mathbf{x}_{\{p+1, \dots, s\}}^*\|_2^2 \leq \int_p^s \frac{1}{x} dx = \log(s-p),$$



and condition (4) reads  $\frac{1}{p+q} > \rho^{2k} \log(s-p)$ . Therefore, Lemma 1 holds for

$$k > \frac{\log(\log(s)) + \log(p+q)}{\log(1/\rho^2)}.$$

Using a partition as given in Lemma 2 gives a sufficient number of iterations

$$\begin{aligned} \bar{n} &= \sum_{i=0}^r k_i = \sum_{i=1}^r \frac{\log(\log(s)) + \log(\sum_{j=1}^i q_j)}{\log(1/\rho^2)} + n_0 \\ &\leq (\log_2(s) + 1) \frac{\log(\log(s)) + \log(2s)}{\log(1/\rho^2)} + n_0. \end{aligned}$$

Again, in this case, the RIC has to be valid at the order  $\Omega(s \cdot s^{\text{polylog}(s)})$  for (GHTP<sup>2</sup>).

**The case**  $0 < \alpha < 1/2$  can be treated in the exact same way by approximating the 2-norm with an integral. This yields, using the same support decomposition,

$$\bar{n} \leq (\log_2(s) + 1) \left( \frac{(1-2\alpha)\log(s)}{\log(1/\rho^2)} + \frac{2\alpha\log(2s)}{\log(1/\rho^2)} - \frac{\log(1-2\alpha)}{\log(1/\rho^2)} \right).$$

Consider **an almost flat  $s$ -sparse vector**  $\mathbf{x}$  such that there exists an  $\varepsilon \geq 0$  with  $1 - \varepsilon \leq x_j^* \leq 1$ , for  $j = 1, \dots, s$  (this corresponds to  $\alpha = 0$ ). In this case, we have that

$$(1 - \varepsilon)^2 (s - p) \leq \|\mathbf{x}_{\{p+1, \dots, s\}^*}\|_2^2 \leq s - p$$

Hence condition (4) now reads  $1 > \rho^{2k} \frac{s-p}{1-\varepsilon}$  and is fulfilled whenever

$$k > \frac{\log\left(\frac{s-p}{1-\varepsilon}\right)}{\log(1/\rho^2)}.$$

Using the decomposition given in Lemma 2, we get that

$$\bar{n} \leq \frac{\log(s/(1-\varepsilon))^2}{\log(1/\rho^2)} + n_0$$

iterations are sufficient to recover the signal  $\mathbf{x}$ . This gives a RIC in the order of  $\Omega(s^{\log(s)})$  when considering (GHTP<sup>2</sup>) for the power vector case. All of the previous results can be summarized in the following corollary:

**Corollary 1** *Let  $\mathbf{x}$  be an  $s$ -sparse vector such that its non-decreasing rearrangement can be written as  $x_j^* = 1/j^\alpha$ , for all  $1 \leq j \leq s$ , for some  $\alpha \geq 0$  or  $1 - \varepsilon \leq x_j^* \leq 1$ , for some  $\varepsilon \geq 0$ . Then for any matrix  $\mathbf{A} \in \mathbb{C}^{m \times N}$ ,  $\mathbf{x}$  can be recovered from  $\mathbf{y} = \mathbf{A}\mathbf{x}$  in at most  $\bar{n} = C \frac{\text{polylog}(s)}{\log(1/\rho^2)} + n_0$  iterations of ( $f$ -HTP) provided that the RIP conditions are satisfied at the order  $\Omega(s + 2f(\text{polylog}(s)))$ . The constant  $C$  and the polynomial involved depend only on  $\alpha$  and  $\rho$*

As a consequence, (GHTP<sup>2</sup>) requires a RIC in the order of  $\Omega(s + 2s^{\log(s)})$ . Similarly, considering  $f(n) = 2n$  yields a RIC in the order of  $\Omega(3s + \text{polylog}(s))$  which

is tractable and still provides a strong speed improvement over the original (GHTP) (even if the complexity remains in the same order, the constant in front is much lower). Some examples are summarized in Table 2.

**Table 2** Examples of RIC-orders that the measurement matrix needs to obey for different ( $f$ -HTP) algorithms (these are just order of magnitude)

$f$	$s$	$n$	$2n$	$n(n+1)/2$	$2^{n-1}$
RIC	$3s$	$3s + 4C\text{polylog}(s)$	$3s + 8C\text{polylog}(s)$	$3s + 4C\text{polylog}^2(s)$	$s + 2s^{\text{polylog}(s)}$
Name	(HTP)	(GHTP)			(GHTP <sup>2</sup> )

## 4 Nonuniform Recovery via ( $f$ -HTP)

We consider here the problem of recovering a particular fixed vector  $\mathbf{x}$  instead of recovering any vector for a given matrix  $\mathbf{A}$ .

### 4.1 Useful Inequalities

We recall here some results regarding the tail distribution of some random variables and the probability distribution of the smallest singular value of a subgaussian matrix [FR13]. These results play an important role in proving the nonuniform recovery of vectors via ( $f$ -HTP).

**Lemma 3 ([FR13])** *Let  $\mathbf{A} \in \mathbb{R}^{m \times N}$  be a subgaussian matrix, the following inequalities hold*

$$\mathbb{P}(\|\mathbf{A}_S^* \mathbf{A}_S - \mathbf{I}\|_{2 \rightarrow 2} > \delta) \leq 2 \exp(-c' \delta^2 m) \quad (6)$$

$$\mathbb{P}(|\langle a_\ell, \mathbf{v} \rangle| > t \|\mathbf{v}\|_2) \leq 4 \exp(-c'' t^2 m), \quad (7)$$

where  $c''$  depends only on the distribution.

### 4.2 Recovery

Following [BFH13, Prop.9], we can see that with high probability the algorithms make no mistakes when selecting the indices. This statement is true while the size of the index set selected at a given iteration is strictly smaller than the actual sparsity of the signal and under a condition on the shape of the vector to be recovered. This result is summarized in the following proposition:

**Proposition 1** Let  $\lambda \geq 1$  and let  $\mathbf{x} \in \mathbb{C}^N$  be an  $s$ -sparse vector such that  $x_1^* \leq \lambda x_s^*$ . If  $A \in \mathbb{R}^{m \times N}$  is a sub-Gaussian matrix with

$$m \geq Cs \ln(N),$$

then with high probability ( $\geq 1 - 2N^{-c}$ ) and for any error vector  $\mathbf{e} \in \mathbb{C}^m$  such that  $\|\mathbf{e}\|_2 \leq \gamma x_s^*$  the sequences  $S^n$  and  $\mathbf{x}^n$  produced by ( $f$ -HTP) with  $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$  satisfy, at iteration  $n_0 - 1$  (where  $n_0$  denotes the smallest integer such that  $f(n) \geq s$ ):

$$S^{n_0-1} \subset S \quad (8)$$

where the constant  $\gamma$  depends only on  $\lambda$  and the constant  $C$  on  $\lambda$  and  $c$ .

**Remark 3** It is worth mentioning that the proof of this Proposition does not apply to the (HTP) algorithm. Indeed, the result holds only while the number of indices is strictly smaller than the actual sparsity. This condition is never met with  $f(n) = s$ .

*Proof.* The proof follows from our previous results. We show that, with high probability,  $S^n \subseteq S$  for all  $1 \leq n \leq n_0 - 1$ . For this we need to show that  $\chi_n > \zeta_n$  where we define

$$\begin{aligned} \chi_n &:= [(\mathbf{x}^{n-1} + \mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^{n-1}))_S]_{f(n)}^*, \\ \zeta_n &:= [(\mathbf{x}^{n-1} + \mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^{n-1}))_{\bar{S}}]_1^*. \end{aligned}$$

Literaly, with  $\mathbf{z}^n := \mathbf{x}^n - \mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^n)$   $\chi_n$  is the  $f(n)^{th}$  largest absolute entry of  $\mathbf{z}^n$  on the support  $S$  of  $\mathbf{x}$  while  $\zeta_n$  is the largest absolute entry of  $\mathbf{z}^n$  on its complement.

$\chi_n > \zeta_n$  for all  $1 \leq n \leq n_0 - 1$  is true with failure probability  $P$ . And we have

$$P := \mathbb{P}(\exists n \in \{1, \dots, n_0 - 1\} : \zeta_n \geq \chi_n \text{ and } (\chi_{n-1} > \zeta_{n-1}, \dots, \chi_1 > \zeta_1)) \quad (9)$$

$$P \leq \mathbb{P}\left(\|\mathbf{A}_{S \cup \{\ell\}}^* \mathbf{A}_{S \cup \{\ell\}} - \mathbf{I}\|_{2 \rightarrow 2} > \delta \text{ for some } \ell \in \bar{S}\right) \quad (10)$$

$$+ \sum_{n=1}^{n_0-1} \mathbb{P}\left(\zeta_n \geq \chi_n, (\chi_{n-1} > \zeta_{n-1}, \dots, \chi_1 > \zeta_1), (\|\mathbf{A}_{S \cup \{\ell\}}^* \mathbf{A}_{S \cup \{\ell\}} - \mathbf{I}\|_{2 \rightarrow 2} \leq \delta \text{ for all } \ell \in \bar{S})\right), \quad (11)$$

Defining  $T^{s-f(n-1)}$  as the set of indices corresponding to the  $s - f(n-1)$  smallest absolute entries of  $\mathbf{z}^n$  on  $S$  we can easily verify that

$$\chi_n \geq \frac{1}{\sqrt{s-f(n-1)}} \left( \|\mathbf{x}_{T^{s-f(n-1)}}\|_2 - \delta \|\mathbf{x} - \mathbf{x}^{n-1}\|_2 - \sqrt{1+\delta} \|\mathbf{e}\|_2 \right).$$

And similarly we have

$$\zeta_n \leq \max_{\ell \in \bar{S}} |\langle a_\ell, \mathbf{A}(\mathbf{x} - \mathbf{x}^{n-1}) \rangle| + \sqrt{1+\delta} \|\mathbf{e}\|_2.$$

Finally with  $\mathbb{P}'(E)$  denoting the probability of an event  $E$  intersected with the event  $\left( (\chi_{n-1} > \zeta_{n-1}, \dots, \chi_1 > \zeta_1), (\|\mathbf{A}_{S \cup \{\ell\}}^* \mathbf{A}_{S \cup \{\ell\}} - \mathbf{I}\|_{2 \rightarrow 2} \leq \delta \text{ for all } \ell \in \bar{S}) \right)$  inequality (11) reads

$$\begin{aligned} & \mathbb{P}'(\zeta_n \geq \chi_n) \\ & \leq \mathbb{P}' \left( \max_{\ell \in \bar{S}} |\langle a_\ell, \mathbf{A}(\mathbf{x} - \mathbf{x}^{n-1}) \rangle| > \frac{1}{\sqrt{s-f(n-1)}} (\|\mathbf{x}_{T^{s-f(n-1)}}\|_2 - \delta \|\mathbf{x} - \mathbf{x}^{n-1}\|_2) - 2\sqrt{1+\delta} \|\mathbf{e}\|_2 \right) \\ & \leq \mathbb{P}' \left( \max_{\ell \in \bar{S}} |\langle a_\ell, \mathbf{A}(\mathbf{x} - \mathbf{x}^{n-1}) \rangle| > \frac{\delta}{\sqrt{s-f(n-1)}} \|\mathbf{x} - \mathbf{x}^{n-1}\|_2 \right) \end{aligned} \quad (12)$$

where the last inequality follows from the fact that:

$$\begin{aligned} & \frac{1}{\sqrt{s-f(n-1)}} (\|\mathbf{x}_{T^{s-f(n-1)}}\|_2 - \delta \|\mathbf{x} - \mathbf{x}^{n-1}\|_2) - 2\sqrt{1+\delta} \|\mathbf{e}\|_2 \\ & \geq \frac{\delta}{\sqrt{s-f(n-1)}} \|\mathbf{x} - \mathbf{x}^{n-1}\|_2 \end{aligned} \quad (13)$$

whenever

$$1 - 2\sqrt{1+\delta} \gamma \geq 2\delta \left( \frac{\lambda}{\sqrt{1-\delta^2}} + \frac{\sqrt{1+\delta}}{1-\delta} \gamma \right). \quad (14)$$

Indeed, inequality (13) is equivalent to

$$\frac{1}{\sqrt{s-f(n-1)}} \|\mathbf{x}_{T^{s-f(n-1)}}\|_2 - 2\sqrt{1+\delta} \|\mathbf{e}\|_2 \geq \frac{2\delta}{\sqrt{s-f(n-1)}} \|\mathbf{x} - \mathbf{x}^{n-1}\|_2 \quad (15)$$

The left-hand side can be estimated by

$$x_s^* - 2\sqrt{1+\delta} \gamma x_s^* = x_s^* (1 - 2\sqrt{1+\delta} \gamma)$$

and the right-hand side is estimated by

$$\begin{aligned} \|\mathbf{x} - \mathbf{x}^{n-1}\|_2 & \leq \frac{1}{\sqrt{1-\delta^2}} \|\mathbf{x}_{S^{n-1}}\|_2 + \frac{1}{1-\delta} \|(\mathbf{A}^* \mathbf{e})_{S^{n-1}}\|_2, \text{ using estimate (1),} \\ & \leq \frac{\sqrt{s-f(n-1)}}{\sqrt{1-\delta^2}} x_1^* + \frac{\sqrt{1+\delta}}{1-\delta} \|\mathbf{e}\|_2, \\ & \leq \frac{\sqrt{s-f(n-1)}}{\sqrt{1-\delta^2}} \lambda x_s^* + \frac{\sqrt{1+\delta}}{1-\delta} \gamma x_s^*, \\ & \leq \sqrt{s-f(n-1)} \left( \frac{\lambda}{\sqrt{1-\delta^2}} + \frac{\sqrt{1+\delta}}{1-\delta} \gamma \right) x_s^*. \end{aligned}$$

Hence condition (14) is verified by choosing  $\delta$  then  $\gamma$  (depending on  $\lambda$ ) small enough.

Finally using the fact that  $\|\mathbf{A}(\mathbf{x} - \mathbf{x}^{n-1})\|_2 \leq \sqrt{1 + \delta} \|\mathbf{x} - \mathbf{x}^{n-1}\|_2$ , the failure probability from Equation (12), can be further approximated by

$$\mathbb{P}'(\zeta_n > \chi_n) \leq \mathbb{P}'\left(\max_{\ell \in \bar{S}} |\langle a_\ell, \mathbf{A}(\mathbf{x} - \mathbf{x}^{n-1}) \rangle| > \frac{\delta}{\sqrt{1 + \delta}} \|\mathbf{A}(\mathbf{x} - \mathbf{x}^{n-1})\|_2\right) \quad (16)$$

Combining these results with Equations (7) and (6), we finally get that

$$P \leq 2(N - s) \exp(-c' \delta^2 m) + 4(N - s)(n_0 - 1) \exp\left(-\frac{c'' \delta^2 m}{(1 + \delta)s}\right).$$

This leads to

$$P \leq 2N^2 \exp\left(-\frac{c''' m}{s}\right)$$

with an appropriate choice of  $c'''_\delta$ .  $\square$

### 4.3 Hybrid Algorithms

We may ask ourselves whether Proposition 1 is of interest or not as it does not lead to the complete recovery of  $\mathbf{x}$ . However, Proposition 1 ensures us that we can create hybrid algorithm with the ( $f$ -HTP) framework where we can make large steps first until a certain criterion is met, and then adaptively reduce the increase of the index set's size until it gets to the sparsity of the signal. Algorithm 1 gives an example of such an hybrid algorithm.

**Data:** A matrix  $\mathbf{A} \in \mathbb{R}^{m \times N}$ , a measurement vector  $\mathbf{y} \in \mathbb{C}^m$ , a switching step  $n \in \mathbb{N}, n \leq n_0$

**Result:** an  $s$  sparse signal  $\mathbf{x}$

Set  $S^0 = \emptyset, \mathbf{x}^0 = \mathbf{0}, nIter = 0$ ;

**while**  $nIter \leq n$  **do**

    Do an iteration of (GHTP<sup>2</sup>);

$nIter = nIter + 1$ ;

**end**

**while** Convergence is not done **do**

    Do an iteration of (GHTP);

$nIter = nIter + 1$ ;

**end**

**Algorithm 1:** Example of an hybrid algorithm for sparse signal recovery

The only important thing to be careful of is that we stay below the sparsity when we start reducing the number of indices added per at each iteration. Moreover, even if Proposition 1 does not ensure convergence of the algorithm until the very last important index, it was shown in [BFH13] that (GHTP) does converge in  $s$  iterations.

This ensures us that such a hybrid algorithm can be used for nonuniform recovery and that it converges in a number of iterations  $\bar{n} \leq s$ .

## 5 Numerical Results

This section validates our theoretical findings by some numerical experiments. Note that all the necessary Matlab files can be found on the author's webpage<sup>2</sup>. Validation is being made with some ( $f$ -HTP) examples compared to the (HTP), (GHTP) and (OMP) algorithms. The following particular index selection functions  $f$  are used:

- $f(n) = s$ : (HTP),
- $f(n) = n$ : (GHTP),
- $f(n) = 2n$ : (GHTP2n),
- $f(n) = n(n+1)/2$ : (GHTP $n^2$ ),
- $f(n) = 2^{n-1}$ : (GHTP $2$ ).

Moreover we will denote by (Hyb1) and (Hyb2) the two algorithms such that the functions  $f$  are defined respectively by

$$f(n) = \begin{cases} 2^{n-1} & , \text{if } n < n_0, \\ 2^{n_0-2} + n - n_0 + 1 & , \text{otherwise,} \end{cases} \quad (\text{Hyb1})$$

$$f(n) = \begin{cases} n(n-1)/2 & , \text{if } n < n_0, \\ (n_0-1)n_0/2 + n - n_0 + 1 & , \text{otherwise.} \end{cases} \quad (\text{Hyb2})$$

The algorithms were tested using 100 randomly generated Gaussian matrices  $\mathbf{A} \in \mathbb{R}^{200 \times 1000}$  each of which were used to recover 10 vectors with randomly generated supports (which represents a total of 1000 random tests for each vector kind and sparsity level). The tests were carried out on three different kinds of vectors to assess the dependence of the algorithms on the decay of the vector  $\mathbf{x}$ ; 'flat' vectors with  $x_j^* = 1$  for  $j \in \{1, \dots, s\}$ , 'linear' vectors with  $x_j^* = (s+1-j)/s$  for  $j \in \{1, \dots, s\}$ , and Gaussian vectors whose  $s$  nonzero entries are independent standard normal random variables.

### 5.1 Successful Recovery and Area of Convergence

We first want to assess the recovery ability of our algorithms by recording the frequency of success as a function of the sparsity. As stopping criterion here we have used the natural one for (HTP) ( $S^n = S^{n-1}$ ) and [ $S \subseteq S^n$  or  $\|\mathbf{x} - \mathbf{x}^n\|_2 / \|\mathbf{x}\|_2 < 10^{-4}$ ]

---

<sup>2</sup> <http://www.math.drexel.edu/~jb3455/publi.html>

for ( $f$ -HTP) and (OMP)<sup>3</sup>. A recovered  $\mathbf{x}$  is recorded as a success whenever the relative error is smaller than  $10^{-4}$ .

As expected, the steeper the index selection function the harder it is for the algorithm to converge. As a consequence (see Fig. 1) (GHTP<sup>2</sup>) performs the worst. However, for reasonable functions  $f$  (up to quadratic functions) the range of convergence of the algorithm is similar to the original one. Moreover, due to the reshuffling of the index set, our family of functions tend to perform better than a classical (OMP).

## 5.2 Number of Iterations for Successful Recovery

One important reason for introducing this generalized family of functions is to lower the number of iterations needed for convergence. Indeed, while the reshuffling of the active set can be seen as an advantage in terms of recovery capability of our algorithms, it takes away any chance of faster implementation, using for instance  $QR$  updates in the inner loop. The following set of graphs (depicted in Figure 2) analyzes the maximum number of iterations needed for recovery.

Three things are worth mentioning. First, as already stated in Remark 2, the maximum number of iterations suggested by Theorem 2 is a very rough overestimation of the actual number of iterations. This is mainly due to the fact that the proof of Theorem 2 relies on the geometric decay of  $\|\mathbf{x}^n - \mathbf{x}\|_2$  that can only be proven for  $n \geq n_0$ . However, as we describe in the next Section, the algorithm picks correct indices much earlier than the  $n_0^{\text{th}}$  iteration. This also shows that the proof Theorem 2 is not optimal as clearly, for most of these algorithms, the RIP suggested is not respected.

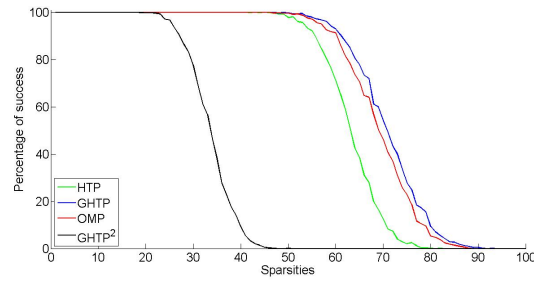
Second, when the algorithms converge, their number of iterations scale according to the underlying function  $f$ . The number of iterations behaves like a logarithm for (GHTP<sup>2</sup>), like a square root for (GHTP $n^2$ ) and linearly for both (GHTP $2n$ ) and (GHTP). Again, (OMP) needs a few more iterations, mainly to compensate the wrong indices that have been picked at an earlier stage of the algorithm.

Finally it is reasonable to think that the analysis carried out in Corollary 1 can be extended to more general vector shapes. However to improve the estimation of the number of iterations we would need to adapt the proof to earlier iterations, instead of starting counting at  $n_0$ .

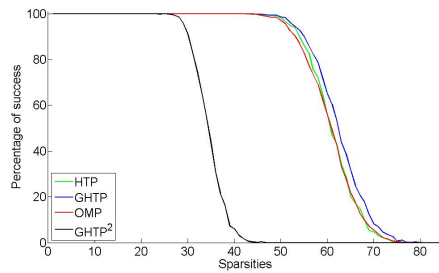
## 5.3 Indices Correctly Captured

We investigate now the ability of our family of algorithms to pick correct indices at each iteration. Figure 3 shows these quantities for the three kinds of vectors (Gaus-

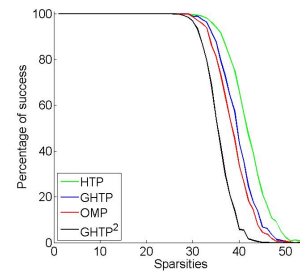
<sup>3</sup> Compared to real applications, we have access here to the true sparsity and the true support of the signal  $\mathbf{x}$ . This stopping criterion needs to be adapted for real-world examples



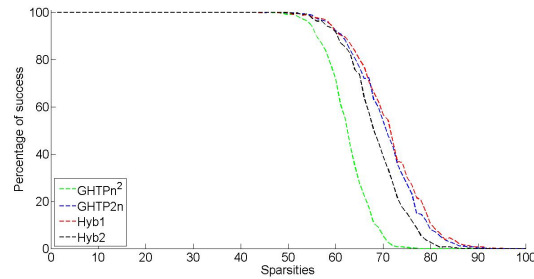
(a) Gaussian vectors - Original algorithms



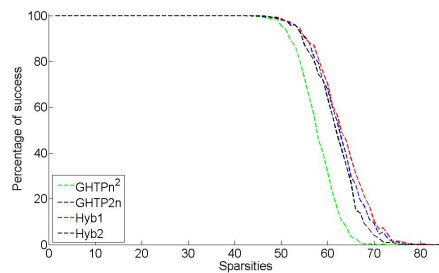
(b) Linear vectors - Original algorithms



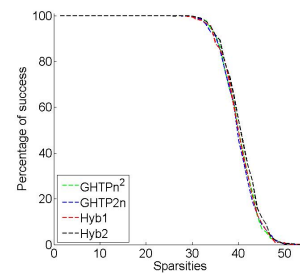
(c) Flat vectors - Original algorithms



(d) Gaussian vectors - Generalized algorithms



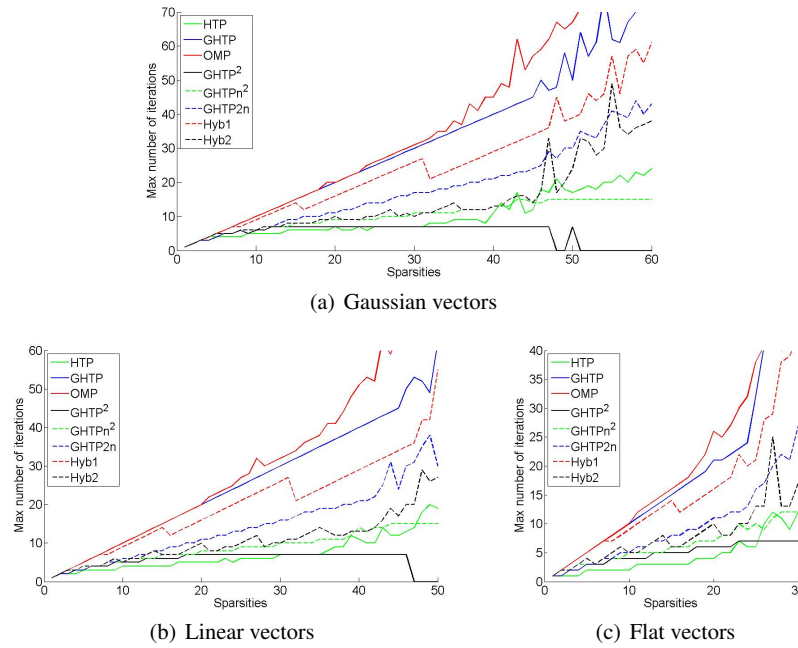
(e) Linear vectors - Generalized algorithms



(f) Flat vectors - Generalized algorithms

**Fig. 1** Frequency of success for the original algorithms ( HTP), (GHTP<sup>2</sup>), (GHTP), and (OMP), two firsts row) and the new generalized approach ((GHTP<sup>2n</sup>), (GHTP<sup>n</sup>), (Hyb1) and (Hyb2), bottom rows) when the original vector is Gaussian (first and third rows), linear (left column) or flat (right column).





**Fig. 2** Maximum number of iterations for exact recovery for the different algorithms when considering Gaussian (top plot), linear (bottom left), or flat (bottom right) vectors.

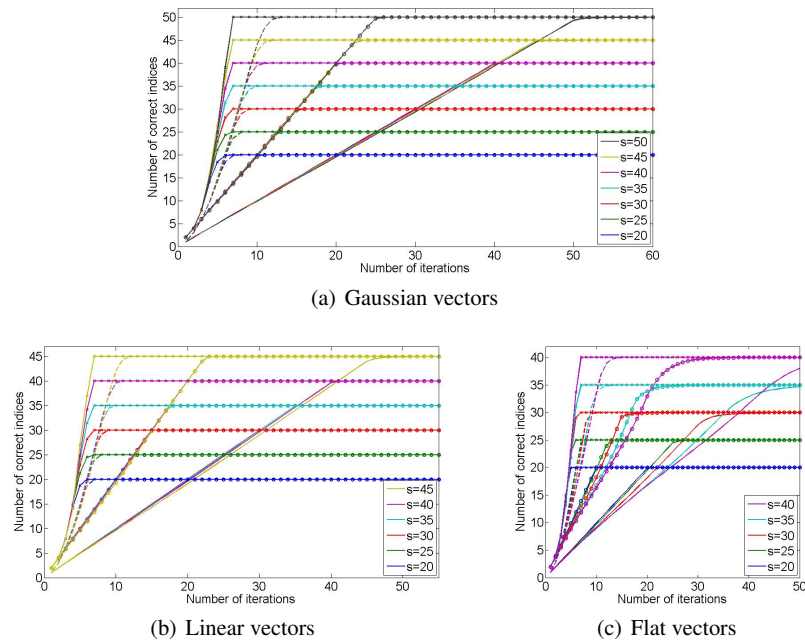
sian to the left, linear in the middle and flat on the right) when dealing with different sparsities and index selection functions (see legend for more details).

As expected most of the algorithms made no mistakes when picking a current active set. This motivates that Proposition 1 can be improved to more general vector shapes.

## 6 Conclusion

This article introduced a class of algorithms that generalizes the Hard Thresholding Pursuit. It allows to overcome both the lack of a priori knowledge regarding the sparsity of the signal to recover and the convergence issue noticed in an earlier extension. We have shown that uniform and nonuniform convergence is possible for all algorithms of this type however sometimes under unrealistic restricted isometry conditions.

Fortunately our numerical results tend to show that the number of iterations implied by our results may be a really rough overestimations. This will lead our future research which would also imply some improved restricted isometry conditions. Moreover, by using a combination of index selecting functions, we are able to pro-



**Fig. 3** Minimum number of correct indices picked at each iteration for different sparsity levels. Continuous lines correspond to (OMP), circles to (GHTP $2n$ ), dashed lines to (GHTP $n^2$ ) and crosses to (GHTP $^2$ ).

duce hybrid algorithms that are both reliable and fast, at least in a nonuniform setting. For such algorithms, a selection of an adequate turning point is needed which is also left for further studies.

## Acknowledgments

The authors want to thank Simon Foucart and Michael Minner for their fruitful comments and suggested literature. The author is also thankful to the NSF for funding his work under the grant number (DMS-1120622).

## References

- [BD09] Thomas Blumensath and Mike E Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27:265–274, 2009.
- [BFH13] Jean-Luc Bouchot, Simon Foucart, and Pawel Hitczenko. Hard thresholding pursuit algorithms: Number of iterations. submitted, 2013.

- [CP11] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision, 40:120–145, 2011.
- [CR05] Emmanuel J. Candès and Justin Romberg.  $\ell_1$ -magic: Recovery of sparse signals via convex programming. URL: [www.acm.caltech.edu/l1magic/downloads/l1magic.pdf](http://www.acm.caltech.edu/l1magic/downloads/l1magic.pdf), 4, 2005.
- [CT05] Emmanuel J. Candès and Terence Tao. Decoding by linear programming. Information Theory, IEEE Transactions on, 51:4203–4215, 2005.
- [CWB08] Emmanuel J Candès, Michael B Wakin, and Stephen P Boyd. Enhancing sparsity by reweighted  $\ell_1$  minimization. Journal of Fourier Analysis and Applications, 14:877–905, 2008.
- [Fou11] Simon Foucart. Hard thresholding pursuit: an algorithm for compressive sensing. SIAM Journal on numerical analysis, 2011.
- [Fou12] Simon Foucart. Sparse recovery algorithms: sufficient conditions in terms of restricted isometry constants. In Approximation Theory XIII: San Antonio 2010. Springer, 2012.
- [FR13] Simon Foucart and Holger Rauhut. A mathematical introduction to compressive sensing. Birkhäuser, 2013.
- [NT09] Deanna Needell and Joel A Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. Applied and Computational Harmonic Analysis, 26:301–321, 2009.
- [TG07] Joel A Tropp and Anna C Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. Information Theory, IEEE Transactions on, 53:4655–4666, 2007.
- [Tro04] Joel A Tropp. Greed is good: Algorithmic results for sparse approximation. Information Theory, IEEE Transactions on, 50:2231–2242, 2004.
- [WN10] David Wipf and Srikantan Nagarajan. Iterative reweighted  $\ell_1$  and  $\ell_2$  methods for finding sparse solutions. IEEE Journal of Selected Topics in Signal Processing, 4:317–329, 2010.
- [YZ11] Junfeng Yang and Yin Zhang. Alternating direction algorithms for  $\ell_{1,1}$ -problems in compressive sensing. SIAM Journal on Scientific Computing, 33:250–278, 2011.
- [Zha11] Tong Zhang. Sparse recovery with orthogonal matching pursuit under RIP. Information Theory, IEEE Transactions on, 57:6215–6221, 2011.