HARD THRESHOLDING PURSUIT: AN ALGORITHM FOR COMPRESSIVE SENSING[*]

SIMON FOUCART[†]

Abstract. We introduce a new iterative algorithm to find sparse solutions of underdetermined linear systems. The algorithm, a simple combination of the Iterative Hard Thresholding algorithm and of the Compressive Sampling Matching Pursuit algorithm, is called Hard Thresholding Pursuit. We study its general convergence, and notice in particular that only a finite number of iterations are required. We then show that, under a certain condition on the restricted isometry constant of the matrix of the linear system, the Hard Thresholding Pursuit algorithm indeed finds all s-sparse solutions. This condition, which reads $\delta_{3s} < 1/\sqrt{3}$, is heuristically better than the sufficient conditions currently available for other Compressive Sensing algorithms. It applies to fast versions of the algorithm, too, including the Iterative Hard Thresholding algorithm. Stability with respect to sparsity defect and robustness with respect to measurement error are also guaranteed under the condition $\delta_{3s} < 1/\sqrt{3}$. We conclude with some numerical experiments to demonstrate the good empirical performance and the low complexity of the Hard Thresholding Pursuit algorithm.

Key words. compressive sensing, sparse recovery, iterative algorithms, thresholding

AMS subject classifications. 65F10, 65J20, 15A29, 94A12

1. Introduction.

In many engineering problems, high-dimensional signals, modeled by vectors in $\mathbb{C}^N$, are observed via low-dimensional measurements, modeled by vectors $\mathbf{y} \in \mathbb{C}^m$ with $m \ll N$, and one wishes to reconstruct the signals from the measurements. Without any prior assumption on the signals, this cannot be hoped for, but the maturing field of Compressive Sensing has shown the feasibility of such a program when the signals are sparse. The objective is then to find suitable measurement matrices $A \in \mathbb{C}^{m \times N}$ and efficient reconstruction algorithms in order to solve underdetermined systems of linear equations $A\mathbf{x} = \mathbf{y}$ when the solutions have only few nonzero components. To date, measurement matrices allowing the reconstruction of s-sparse vectors — vectors with at most s nonzero components — with the minimal number $m \approx c\, s\, \ln(N/s)$ of measurements have only been constructed probabilistically. As for the reconstruction procedure, a very popular strategy consists in solving the following ℓ_1-minimization problem, known as Basis Pursuit,

$$\underset{\mathbf{z} \in \mathbb{C}^N}{\text{minimize}} \ \|\mathbf{z}\|_1 \qquad \text{subject to} \quad A\mathbf{z} = \mathbf{y}. \tag{BP}$$

The goal of this paper is to propose an alternative strategy that combines two existing iterative algorithms. The new algorithm, called Hard Thresholding Pursuit, together with some variants, is formally introduced in Section 2 after a few intuitive justifications. In Section 3, we analyze the theoretical performance of the algorithm. In particular, we show that the Hard Thresholding Pursuit algorithm allows stable and robust reconstruction of sparse vectors if the measurement matrix satisfies some restricted isometry conditions that, heuristically, are the best available so far. Finally, the numerical experiments of Section 4 show that the algorithm also performs well in practice.

[*] This work was initiated when the author was with the Université Pierre et Marie Curie in Paris. It was supported by the French National Research Agency (ANR) through the project ECHANGE (ANR-08-EMER-006).

[†] Drexel University, Philadelphia

2. The algorithm's rationale. In this section, we outline two families of iterative algorithms for Compressive Sensing. We then combine their premises to create the new family of Hard Thresholding Pursuit algorithms.

2.1. Iterative Hard Thresholding. The Iterative Hard Thresholding (IHT) algorithm was first introduced for sparse recovery problems by Blumensath and Davies in [2]. Elementary analyses, in particular the one in [11], show the good theoretical guarantees of this algorithm. It is built on the simple intuitions: that solving the rectangular system $A\mathbf{x} = \mathbf{y}$ amounts to solving the square system $A^*A\mathbf{x} = A^*\mathbf{y}$; that classical iterative methods suggest to define a sequence $(\mathbf{x}^n)$ by the recursion $\mathbf{x}^{n+1} = (I - A^*A)\mathbf{x}^n + A^*\mathbf{y}$; and that, since sparse vectors are desired, each step should involve the hard thresholding operator H_s that keeps s largest (in modulus) components of a vector and sets the other ones to zero (in case s largest components are not uniquely defined, we select the smallest possible indices). This is all the more justified by the fact that the restricted isometry property — see Section 3 — ensures that the matrix A^*A behaves like the identity when its domain and range are restricted to small support sets. Thus, the contributions to $\mathbf{x}^{n+1}$ of the terms $(I - A^*A)\mathbf{x}^n$ and $A^*\mathbf{y}$ are roughly $\mathbf{x}^n - \mathbf{x}^n = 0$ and $\mathbf{x}$, so that $\mathbf{x}^{n+1}$ appears as a good approximation to $\mathbf{x}$. This yields the following algorithm, whose inputs — like all algorithms below — are the measurement vector $\mathbf{y}$, the measurement matrix A, and the anticipated sparsity s (the prior estimation of s is a drawback of such algorithms compared with Basis Pursuit, but the latter generally requires a prior estimation of the measurement error). One may also prescribe the number of iterations.
Start with an s-sparse $\mathbf{x}^0 \in \mathbb{C}^N$, typically $\mathbf{x}^0 = 0$, and iterate the scheme

$$\mathbf{x}^{n+1} = H_s(\mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n)) \tag{IHT}$$

until a stopping criterion is met.
One may be slightly more general and consider an algorithm (IHT$^\mu$) by allowing a factor $\mu \neq 1$ in front of $A^*(\mathbf{y} - A\mathbf{x}^n)$ — this was called Gradient Descent with Sparsification (GDS) in [12].
Start with an s-sparse $\mathbf{x}^0 \in \mathbb{C}^N$, typically $\mathbf{x}^0 = 0$, and iterate the scheme

$$\mathbf{x}^{n+1} = H_s(\mathbf{x}^n + \mu A^*(\mathbf{y} - A\mathbf{x}^n)) \tag{IHT$^\mu$}$$

until a stopping criterion is met.
One may even allow the factor μ to depend on the iteration, hence considering the Normalized Iterative Hard Thresholding (NIHT) algorithm described as follows.
Start with an s-sparse $\mathbf{x}^0 \in \mathbb{C}^N$, typically $\mathbf{x}^0 = 0$, and iterate the scheme

$$\mathbf{x}^{n+1} = H_s(\mathbf{x}^n + \mu_n A^*(\mathbf{y} - A\mathbf{x}^n)) \tag{NIHT}$$

until a stopping criterion is met.
The original terminology of Normalized Iterative Hard Thresholding used in [4] corresponds to the specific choice (where the notation $\mathbf{z}_T \in \mathbb{C}^N$ stands for a vector equal to a vector $\mathbf{z} \in \mathbb{C}^N$ on a set T and to zero outside of T)

$$\mu_n = \frac{\|\big(A^*(\mathbf{y} - A\mathbf{x}^n)\big)_{S^n}\|_2^2}{\|A\big(\big(A^*(\mathbf{y} - A\mathbf{x}^n)\big)_{S^n}\big)\|_2^2}, \qquad S^n := \operatorname{supp}(\mathbf{x}^n),$$

unless, fixing a prescribed $0 < \eta < 1$, one has $\mu_n > \eta \|\mathbf{x}^n - \mathbf{x}^{n+1}\|_2^2 / \|A(\mathbf{x}^n - \mathbf{x}^{n+1})\|_2^2$ — the reasons for this will become apparent in (3.2). In this case, the factor μ_n is halved until the exception vanishes.

2.2. Compressive Sensing Matching Pursuit. The Compressive Sampling Matching Pursuit (CoSaMP) algorithm proposed by Needell and Tropp in [18] and the Subspace Pursuit (SP) algorithm proposed by Dai and Milenkovic in [8] do not provide better theoretical guarantees than the simple IHT algorithm, but they do offer better empirical performances. These two algorithms, grouped in a family called Compressive Sensing Matching Pursuit (CSMP) for convenience here, were devised to enhance the Orthogonal Matching Pursuit (OMP) algorithm initially proposed in [17]. The basic idea consists in chasing a good candidate for the support, and then finding the vector with this support that best fits the measurements. For instance, the CoSaMP algorithm reads as follows.
Start with an s-sparse $\mathbf{x}^0 \in \mathbb{C}^N$, typically $\mathbf{x}^0 = 0$, and iterate the scheme

$$U^{n+1} = \mathrm{supp}(\mathbf{x}^n) \cup \big\{\text{indices of } 2s \text{ largest entries of } A^*(\mathbf{y} - A\mathbf{x}^n)\big\}, \qquad (\mathrm{CoSaMP}_1)$$
$$\mathbf{u}^{n+1} = \mathrm{argmin}\big\{\|\mathbf{y} - A\mathbf{z}\|_2, \mathrm{supp}(\mathbf{z}) \subseteq U^{n+1}\big\}, \qquad (\mathrm{CoSaMP}_2)$$
$$\mathbf{x}^{n+1} = H_s(\mathbf{u}^{n+1}), \qquad (\mathrm{CoSaMP}_3)$$

until a stopping criterion is met. The SP algorithm is very similar: s replaces $2s$ in (CoSaMP$_1$), and the two steps $S^{n+1} = \{\text{indices of } s \text{ largest entries of } \mathbf{u}^{n+1}\}$ and $\mathbf{x}^{n+1} = \mathrm{argmin}\{\|\mathbf{y} - A\mathbf{z}\|_2, \mathrm{supp}(\mathbf{z}) \subseteq S^{n+1}\}$ replace (CoSaMP$_3$). In a sense, the common approach in the CSMP family defies intuition, because the candidate for the support uses the largest components of the vector $A^*A(\mathbf{x} - \mathbf{x}^n) \approx \mathbf{x} - \mathbf{x}^n$, and not of a vector close to $\mathbf{x}$.

2.3. Hard Thresholding Pursuit. Sticking to the basic idea of chasing a good candidate for the support then finding the vector with this support that best fits the measurements, but inspired by intuition from the IHT algorithm, it seems natural to select instead the s largest components of $\mathbf{x}^n + A^*A(\mathbf{x} - \mathbf{x}^n) \approx \mathbf{x}$. This combination of the IHT and CSMP algorithms leads to the Hard Thresholding Pursuit (HTP) algorithm described as follows.
Start with an s-sparse $\mathbf{x}^0 \in \mathbb{C}^N$, typically $\mathbf{x}^0 = 0$, and iterate the scheme

$$S^{n+1} = \big\{\text{indices of } s \text{ largest entries of } \mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n)\big\}, \qquad (\mathrm{HTP}_1)$$
$$\mathbf{x}^{n+1} = \mathrm{argmin}\big\{\|\mathbf{y} - A\mathbf{z}\|_2, \mathrm{supp}(\mathbf{z}) \subseteq S^{n+1}\big\}, \qquad (\mathrm{HTP}_2)$$

until a stopping criterion is met. A natural criterion here is $S^{n+1} = S^n$, since then $\mathbf{x}^k = \mathbf{x}^n$ for all $k \geq n$, although there is no guarantee that this should occur. Step (HTP$_2$), often referred to as debiasing, has been shown to improve performances in other algorithms, too. As with the Iterative Hard Thresholding algorithms, we may be more general and consider an algorithm (HTP$^\mu$) by allowing a factor $\mu \neq 1$ as follows.
Start with an s-sparse $\mathbf{x}^0 \in \mathbb{C}^N$, typically $\mathbf{x}^0 = 0$, and iterate the scheme

$$S^{n+1} = \big\{\text{indices of } s \text{ largest entries of } \mathbf{x}^n + \mu A^*(\mathbf{y} - A\mathbf{x}^n)\big\}, \qquad (\mathrm{HTP}_1^\mu)$$
$$\mathbf{x}^{n+1} = \mathrm{argmin}\big\{\|\mathbf{y} - A\mathbf{z}\|_2, \mathrm{supp}(\mathbf{z}) \subseteq S^{n+1}\big\}, \qquad (\mathrm{HTP}_2^\mu)$$

until a stopping criterion is met.

By allowing the factor μ to depend on the iteration according to the specific choice

$$\mu_n = \frac{\|\big(A^*(\mathbf{y} - A\mathbf{x}^n)\big)_{S^n}\|_2^2}{\|A\big(\big(A^*(\mathbf{y} - A\mathbf{x}^n)\big)_{S^n}\big)\|_2^2},$$

we may also consider the Normalized Hard Thresholding Pursuit algorithm described as follows.
Start with an s-sparse $\mathbf{x}^0 \in \mathbb{C}^N$, typically $\mathbf{x}^0 = 0$, and iterate the scheme

$$S^{n+1} = \big\{\text{indices of } s \text{ largest entries of } \mathbf{x}^n + \mu_n A^*(\mathbf{y} - A\mathbf{x}^n)\big\}, \tag{NHTP$_1$}$$
$$\mathbf{x}^{n+1} = \operatorname{argmin}\big\{\|\mathbf{y} - A\mathbf{z}\|_2, \operatorname{supp}(\mathbf{z}) \subseteq S^{n+1}\big\}, \tag{NHTP$_2$}$$

until a stopping criterion is met.

In all the above algorithms, the second steps require to solve the $s \times s$ system of normal equations $A^*_{S^{n+1}} A_{S^{n+1}} \mathbf{x}^{n+1} = A^*_{S^{n+1}} \mathbf{y}$. If these steps are judged too costly, we may consider instead a fast version of the Hard Thresholding Pursuit algorithms, where the orthogonal projection is replaced by a certain number k of gradient descent iterations. This leads for instance to the algorithm (FHTP$^\mu$) described below. In the special case $\mu = 1$, we call the algorithm Fast Hard Thresholding Pursuit (FHTP) — note that $k = 0$ corresponds to the classical IHT algorithm and that $k = \infty$ corresponds to the HTP algorithm.
Start with an s-sparse $\mathbf{x}^0 \in \mathbb{C}^N$, typically $\mathbf{x}^0 = 0$, and iterate the scheme

$$S^{n+1} = \operatorname{supp}(\mathbf{u}^{n+1,1}), \qquad \mathbf{u}^{n+1,1} := H_s(\mathbf{x}^n + \mu A^*(\mathbf{y} - A\mathbf{x}^n)), \tag{FHTP$^\mu_1$}$$
$$\mathbf{x}^{n+1} = \mathbf{u}^{n+1,k+1}, \qquad \mathbf{u}^{n+1,\ell+1} := \big(\mathbf{u}^{n+1,\ell} + t_{n+1,\ell} A^*(\mathbf{y} - A\mathbf{u}^{n+1,\ell})\big)_{S^{n+1}}, \tag{FHTP$^\mu_2$}$$

until a stopping criterion is met.
A simple choice for $t_{n+1,\ell}$ is simply $t_{n+1,\ell} = 1$, while a wiser choice, corresponding to a steepest descent, is

$$t_{n+1,\ell} = \frac{\|\big(A^*(\mathbf{y} - A\mathbf{u}^{n+1,\ell})\big)_{S^{n+1}}\|_2^2}{\|A\big(\big(A^*(\mathbf{y} - A\mathbf{u}^{n+1,\ell})\big)_{S^{n+1}}\big)\|_2^2}. \tag{2.1}$$

3. Theoretical justification. In this section, we analyze the theoretical performances of the proposed algorithms. We first show the convergence of the algorithms under some conditions on the measurement matrix A, precisely on its operator norm then on its restricted isometry constants, which are introduced along the way. Next, we study the exact recovery of sparse vectors as outputs of the proposed algorithms using perfect measurements. Sufficient conditions for successful recovery are given in terms of restricted isometry constants, and we heuristically argue that these conditions are the best available so far. Finally, we prove that these sufficient conditions also guarantee a stable and robust recovery with respect to sparsity defect and to measurement error.

3.1. Convergence. First and foremost, we make a simple observation about the HTP algorithm — or HTP$^\mu$ and NHTP, for that matter. Namely, since there is only a finite number of subsets of $\{1, \ldots, N\}$ with size s, there exist integers $n, p \geq 1$ such that $S^{n+p} = S^n$, so that (HTP$_2$) and (HTP$_1$) yield $\mathbf{x}^{n+p} = \mathbf{x}^n$ and $S^{n+p+1} = S^{n+1}$, and so on until $\mathbf{x}^{n+2p} = \mathbf{x}^{n+p}$ and $S^{n+2p} = S^{n+p} = S^n$. Thus, one actually shows recursively that $\mathbf{x}^{n+kp+r} = \mathbf{x}^{n+r}$ for all $k \geq 1$ and $1 \leq r \leq p$. Simply stated, this takes the following form.

LEMMA 3.1. *The sequences defined by (HTP), (HTP$^\mu$), and (NHTP) are eventually periodic.*

The importance of this observation lies in the fact that, as soon as the convergence of one of these algorithms is established, then we can certify that the limit is exactly achieved after a finite number of iterations. For instance, we establish below the convergence of the HTP algorithm under a condition on the operator norm $\|A\|_{2\to 2} := \sup_{\mathbf{x}\neq 0} \|A\mathbf{x}\|_2/\|\mathbf{x}\|_2$ of the matrix A. This parallels a result of [2], where the convergence of (IHT) was also proved under the condition $\|A\|_{2\to 2} < 1$. Our proof uses the same strategy, based on the decrease along the iterations of the quantity $\|\mathbf{y} - A\mathbf{x}^n\|_2$ (the 'cost'; note that the auxiliary 'surrogate cost' is not mentioned here).

PROPOSITION 3.2. *The sequence* $(\mathbf{x}^n)$ *defined by (HTP*$^\mu$*) converges in a finite number of iterations provided* $\mu\|A\|_{2\to 2}^2 < 1$.

Proof. Let us consider the vector supported on S^{n+1} defined by

$$\mathbf{u}^{n+1} := H_s(\mathbf{x}^n + \mu A^*(\mathbf{y} - A\mathbf{x}^n)).$$

According to the definition of $A\mathbf{x}^{n+1}$, we have $\|\mathbf{y} - A\mathbf{x}^{n+1}\|_2^2 \le \|\mathbf{y} - A\mathbf{u}^{n+1}\|_2^2$, and it follows that

$$\begin{aligned}\|\mathbf{y}-A\mathbf{x}^{n+1}\|_2^2 - \|\mathbf{y}-A\mathbf{x}^n\|_2^2 &\le \|\mathbf{y}-A\mathbf{u}^{n+1}\|_2^2 - \|\mathbf{y}-A\mathbf{x}^n\|_2^2\\ &= \|A(\mathbf{x}^n-\mathbf{u}^{n+1}) + \mathbf{y}-A\mathbf{x}^n\|_2^2 - \|\mathbf{y}-A\mathbf{x}^n\|_2^2\\ &= \|A(\mathbf{x}^n-\mathbf{u}^{n+1})\|_2^2 + 2\Re\langle A(\mathbf{x}^n-\mathbf{u}^{n+1}), \mathbf{y}-A\mathbf{x}^n\rangle. \quad (3.1)\end{aligned}$$

We now observe that $\mathbf{u}^{n+1}$ is a better s-term approximation to $\mathbf{x}^n + \mu A^*(\mathbf{y} - A\mathbf{x}^n)$ than $\mathbf{x}^n$ is, so that

$$\|\mathbf{x}^n + \mu A^*(\mathbf{y} - A\mathbf{x}^n) - \mathbf{u}^{n+1}\|_2^2 \le \|\mu A^*(\mathbf{y} - A\mathbf{x}^n)\|_2^2.$$

After expanding the squares, we obtain

$$2\mu\Re\langle \mathbf{x}^n - \mathbf{u}^{n+1}, A^*(\mathbf{y} - A\mathbf{x}^n)\rangle \le -\|\mathbf{x}^n - \mathbf{u}^{n+1}\|_2^2.$$

Substituting this into (3.1), we derive

$$\|\mathbf{y} - A\mathbf{x}^{n+1}\|_2^2 - \|\mathbf{y} - A\mathbf{x}^n\|_2^2 \le \|A(\mathbf{x}^n - \mathbf{u}^{n+1})\|_2^2 - \frac{1}{\mu}\|\mathbf{x}^n - \mathbf{u}^{n+1}\|_2^2. \quad (3.2)$$

We use the simple inequality

$$\|A(\mathbf{x}^n - \mathbf{u}^{n+1})\|_2^2 \le \|A\|_{2\to 2}^2\, \|\mathbf{x}^n - \mathbf{u}^{n+1}\|_2^2, \quad (3.3)$$

and the hypothesis that $\|A\|_{2\to 2}^2 < 1/\mu$ to deduce that

$$\|\mathbf{y} - A\mathbf{x}^{n+1}\|_2^2 - \|\mathbf{y} - A\mathbf{x}^n\|_2^2 \le -c\, \|\mathbf{x}^n - \mathbf{u}^{n+1}\|_2^2, \quad (3.4)$$

where $c := 1/\mu - \|A\|_{2\to 2}^2$ is a positive constant. This proves that the nonnegative sequence $(\|\mathbf{y} - A\mathbf{x}^n\|_2)$ is nonincreasing, hence it is convergent. Since it is also eventually periodic, it must be eventually constant. In view of (3.4), we deduce that $\mathbf{u}^{n+1} = \mathbf{x}^n$, and in particular that $S^{n+1} = S^n$, for n large enough. This implies that $\mathbf{x}^{n+1} = \mathbf{x}^n$ for n large enough, which implies the required result. □

As seen in (3.3), it is not really the norm of A that matters, but rather its 'norm on sparse vectors'. This point motivates the introduction of the sth order restricted isometry constant $\delta_s = \delta_s(A)$ of a matrix $A \in \mathbb{C}^{m\times N}$. We recall that these were defined in [6] as the smallest $\delta \ge 0$ such that

$$(1-\delta)\|\mathbf{x}\|_2^2 \le \|A\mathbf{x}\|_2^2 \le (1+\delta)\|\mathbf{x}\|_2^2 \qquad \text{for all } s\text{-sparse vectors } \mathbf{x} \in \mathbb{C}^N. \quad (3.5)$$

Replacing (3.3) by $\|A(\mathbf{x}^n - \mathbf{u}^{n+1})\|_2^2 \le (1+\delta_{2s})\,\|\mathbf{x}^n - \mathbf{u}^{n+1}\|_2^2$ in the previous proof immediately yields the following result.

THEOREM 3.3. *The sequence* $(\mathbf{x}^n)$ *defined by (HTP*$^\mu$*) converges in a finite number of iterations provided* $\mu(1+\delta_{2s}) < 1$.

We close this subsection with analogs of Proposition 3.2 and Theorem 3.3 for the fast versions of the Hard Thresholding Pursuit algorithm.

THEOREM 3.4. *For any* $k \ge 0$*, with* $\mu \ge 1/2$ *and with* $t_{n+1,\ell}$ *equal to* 1 *or given by* (2.1)*, the sequence* $(\mathbf{x}^n)$ *defined by (FTHP*$^\mu$*) converges provided* $\mu\|A\|_{2\to 2}^2 < 1$ *or* $\mu(1+\delta_{2s}) < 1$.

Proof. Keeping in mind the proof of Proposition 3.2, we see that it is enough to establish the inequality $\|\mathbf{y} - A\mathbf{x}^{n+1}\|_2 \le \|\mathbf{y} - A\mathbf{u}^{n+1}\|_2$. Since $\mathbf{x}^{n+1} = \mathbf{u}^{n+1,k+1}$ and $\mathbf{u}^{n+1} = \mathbf{u}^{n+1,1}$, we just need to prove that, for any $1 \le \ell \le k$,

$$\|\mathbf{y} - A\mathbf{u}^{n+1,\ell+1}\|_2 \le \|\mathbf{y} - A\mathbf{u}^{n+1,\ell}\|_2. \tag{3.6}$$

Let $A_{S^{n+1}}$ denote the submatrix of A obtained by keeping the columns indexed by S^{n+1} and let $\mathbf{v}^{n+1,\ell+1}, \mathbf{v}^{n+1,\ell} \in \mathbb{C}^s$ denote the subvectors of $\mathbf{u}^{n+1,\ell+1}, \mathbf{u}^{n+1,\ell} \in \mathbb{C}^N$ obtained by keeping the entries indexed by S^{n+1}. With $t_{n+1,\ell} = 1$, we have

$$\begin{aligned}\|\mathbf{y} - A\mathbf{u}^{n+1,\ell+1}\|_2 &= \|\mathbf{y} - A_{S^{n+1}}\mathbf{v}^{n+1,\ell+1}\|_2 = \|(I - A_{S^{n+1}}A_{S^{n+1}}^*)(\mathbf{y} - A_{S^{n+1}}\mathbf{v}^{n+1,\ell})\|_2\\ &\le \|\mathbf{y} - A_{S^{n+1}}\mathbf{v}^{n+1,\ell}\|_2 = \|\mathbf{y} - A\mathbf{u}^{n+1,\ell}\|_2,\end{aligned}$$

where the inequality is justified because the hermitian matrix $A_{S^{n+1}}A_{S^{n+1}}^*$ has eigenvalues in $[0, \|A\|_{2\to 2}^2]$ or $[0, 1+\delta_{2s}]$, hence in $[0, 1/\mu] \subseteq [0,2]$. Thus (3.6) holds with $t_{n+1,\ell} = 1$. With $t_{n+1,\ell}$ given by (2.1), it holds because this is actually the value that minimizes over $t = t_{n+1,\ell}$ the quadratic expression

$$\|\mathbf{y} - A_{S^{n+1}}\mathbf{v}^{n+1,\ell+1}\|_2^2 = \|\mathbf{y} - A_{S^{n+1}}\mathbf{v}^{n+1,\ell} - t\,A_{S^{n+1}}A_{S^{n+1}}^*(\mathbf{y} - A_{S^{n+1}}\mathbf{v}^{n+1,\ell+1})\|_2^2,$$

as one can easily verify. □

3.2. Exact recovery of sparse vectors from accurate measurements.

We place ourselves in the ideal case where the vectors to be recovered are exactly sparse and are measured with infinite precision. Although the main result of this subsection, namely Theorem 3.5, is a particular instance of Theorem 3.8, we isolate it because its proof is especially elegant in this simple case and sheds light on the more involved proof of Theorem 3.8. Theorem 3.5 guarantees the recovery of s-sparse vectors via Hard Thresholding Pursuit under a condition on the $3s$th restricted isometry constant of the measurement matrix. Sufficient conditions of this kind, which often read $\delta_t < \delta_*$ for some integer t related to s and for some specific value δ_*, have become a benchmark for theoretical investigations, because they are pertinent in the analysis a wide range of algorithms. Note that the condition $\delta_t < \delta_*$ is mainly known to be satisfied for random matrices provided the number of measurements scales like

$$m \approx c\,\frac{t}{\delta_*^2}\,\ln(N/t).$$

In fact, the condition $\delta_t < \delta_*$ can only be fulfilled if $m \ge c\,t/\delta_*^2$ — see Appendix for a precise statement and its proof. Since we want to make as few measurements as possible, we may heuristically assess a sufficient condition by the smallness of the ratio t/δ_*^2. In this respect, the sufficient condition $\delta_{3s} < 1/\sqrt{3}$ of this paper

— valid not only for HTP but also for FTHP (in particular for IHT, too) — is currently the best available, as shown in the following table (we did not include the sufficient conditions $\delta_{2s} < 0.473$ and $\delta_{3s} < 0.535$ of [10] and [5], giving the ratios 8.924 and 10.44, because they are only valid for large s). More careful investigations should be carried out in the framework of phase transition, see e.g. [9].

Algorithm	IHT	GDS	CoSaMP	(F)HTP	BP
Reference	[11]	[12]	[11]	this paper	[10]
$\delta_t < \delta_*$	$\delta_{3s} < 0.5$	$\delta_{2s} < 0.333$	$\delta_{4s} < 0.384$	$\delta_{3s} < 0.577$	$\delta_{2s} < 0.465$
Ratio t/δ_*^2	12	18	27.08	**9**	9.243

Before turning to the main result of this subsection, we point out a less common, but sometimes preferable, expression of the restricted isometry constant, i.e.,

$$\delta_s = \max_{|S| \le s} \|A_S^* A_S - \mathrm{Id}\|_{2 \to 2},$$

where A_S denotes the submatrix of A obtained by keeping the columns indexed by S. This enables to observe easily that

$$|\langle \mathbf{u}, (\mathrm{Id} - A^*A)\mathbf{v} \rangle| \le \delta_t \|\mathbf{u}\|_2 \|\mathbf{v}\|_2 \qquad \text{whenever } |\mathrm{supp}(\mathbf{u}) \cup \mathrm{supp}(\mathbf{v})| \le t. \tag{3.7}$$

Indeed, setting $T := \mathrm{supp}(\mathbf{u}) \cup \mathrm{supp}(\mathbf{v})$ and denoting by $\mathbf{u}_T$ and $\mathbf{v}_T$ the subvectors of $\mathbf{u}$ and $\mathbf{v}$ obtained by only keeping the components indexed by T (this notation is in slight conflict with the one used elsewhere in the paper, where $\mathbf{u}_T$ and $\mathbf{v}_T$ would be vectors in $\mathbb{C}^N$), we have

$$\begin{aligned} |\langle \mathbf{u}, (\mathrm{Id} - A^*A)\mathbf{v} \rangle| &= |\langle \mathbf{u}, \mathbf{v} \rangle - \langle A\mathbf{u}, A\mathbf{v} \rangle| = |\langle \mathbf{u}_T, \mathbf{v}_T \rangle - \langle A_T \mathbf{u}_T, A_T \mathbf{v}_T \rangle| \\ &= |\langle \mathbf{u}_T, (\mathrm{Id} - A_T^* A_T)\mathbf{v}_T| \le \|\mathbf{u}_T\|_2 \, \|(\mathrm{Id} - A_T^* A_T)\mathbf{v}_T\|_2 \\ &\le \|\mathbf{u}_T\|_2 \, \|\mathrm{Id} - A_T^* A_T\|_{2\to 2} \|\mathbf{v}_T\|_2 \le \|\mathbf{u}_T\|_2 \, \delta_t \|\mathbf{v}_T\|_2 = \delta_t \|\mathbf{u}\|_2 \|\mathbf{v}\|_2. \end{aligned}$$

It also enables us to observe easily that

$$\|((\mathrm{Id} - A^*A)\mathbf{v})_U\|_2 \le \delta_t \|\mathbf{v}\|_2 \qquad \text{whenever } |U \cup \mathrm{supp}(\mathbf{v})| \le t. \tag{3.8}$$

Indeed, using (3.7), we have

$$\|((\mathrm{Id} - A^*A)\mathbf{v})_U\|_2^2 = \langle ((\mathrm{Id} - A^*A)\mathbf{v})_U, (\mathrm{Id} - A^*A)\mathbf{v} \rangle \le \delta_t \|((\mathrm{Id} - A^*A)\mathbf{v})_U\|_2 \|\mathbf{v}\|_2,$$

and it remains to simplify by $\|((\mathrm{Id} - A^*A)\mathbf{v})_U\|_2$ to obtain (3.8).

THEOREM 3.5. *Suppose that the 3sth order restricted isometry constant of the measurement matrix $A \in \mathbb{C}^{m \times N}$ satisfies*

$$\delta_{3s} < \frac{1}{\sqrt{3}} \approx 0.57735.$$

Then, for any s-sparse $\mathbf{x} \in \mathbb{C}^N$, the sequence $(\mathbf{x}^n)$ defined by (HTP) with $\mathbf{y} = A\mathbf{x}$ converges towards $\mathbf{x}$ at a geometric rate given by

$$\|\mathbf{x}^n - \mathbf{x}\|_2 \le \rho^n \, \|\mathbf{x}^0 - \mathbf{x}\|_2, \qquad \rho := \sqrt{\frac{2\delta_{3s}^2}{1 - \delta_{2s}^2}} < 1. \tag{3.9}$$

Proof. The first step of the proof is a consequence of (HTP$_2$). We notice that $A\mathbf{x}^{n+1}$ is the best ℓ_2-approximation to $\mathbf{y}$ from the space $\{A\mathbf{z}, \mathrm{supp}(\mathbf{z}) \subseteq S^{n+1}\}$, hence it is characterized by the orthogonality condition

$$\langle A\mathbf{x}^{n+1} - \mathbf{y}, A\mathbf{z} \rangle = 0 \qquad \text{whenever } \mathrm{supp}(\mathbf{z}) \subseteq S^{n+1}. \tag{3.10}$$

Since $\mathbf{y} = A\mathbf{x}$, this may be rewritten as

$$\langle \mathbf{x}^{n+1} - \mathbf{x}, A^*A\mathbf{z} \rangle = 0 \qquad \text{whenever } \mathrm{supp}(\mathbf{z}) \subseteq S^{n+1}.$$

We derive in particular

$$\begin{aligned}
\|(\mathbf{x}^{n+1} - \mathbf{x})_{S^{n+1}}\|_2^2 &= \langle \mathbf{x}^{n+1} - \mathbf{x}, (\mathbf{x}^{n+1} - \mathbf{x})_{S^{n+1}} \rangle \\
&= \langle \mathbf{x}^{n+1} - \mathbf{x}, (I - A^*A)\big((\mathbf{x}^{n+1} - \mathbf{x})_{S^{n+1}}\big) \rangle \\
&\underset{(3.7)}{\le} \delta_{2s} \|\mathbf{x}^{n+1} - \mathbf{x}\|_2 \|(\mathbf{x}^{n+1} - \mathbf{x})_{S^{n+1}}\|_2.
\end{aligned}$$

After simplification, we have $\|(\mathbf{x}^{n+1} - \mathbf{x})_{S^{n+1}}\|_2 \le \delta_{2s}\|\mathbf{x}^{n+1} - \mathbf{x}\|_2$. It follows that

$$\begin{aligned}
\|\mathbf{x}^{n+1} - \mathbf{x}\|_2^2 &= \|(\mathbf{x}^{n+1} - \mathbf{x})_{\overline{S^{n+1}}}\|_2^2 + \|(\mathbf{x}^{n+1} - \mathbf{x})_{S^{n+1}}\|_2^2 \\
&\le \|(\mathbf{x}^{n+1} - \mathbf{x})_{\overline{S^{n+1}}}\|_2^2 + \delta_{2s}^2 \|\mathbf{x}^{n+1} - \mathbf{x}\|_2^2.
\end{aligned}$$

After a rearrangement, we obtain

$$\|\mathbf{x}^{n+1} - \mathbf{x}\|_2^2 \le \frac{1}{1 - \delta_{2s}^2} \|(\mathbf{x}^{n+1} - \mathbf{x})_{\overline{S^{n+1}}}\|_2^2. \tag{3.11}$$

The second step of the proof is a consequence of (HTP$_1$). With $S := \mathrm{supp}(\mathbf{x})$, we notice that

$$\|(\mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n))_S\|_2^2 \le \|(\mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n))_{S^{n+1}}\|_2^2.$$

Eliminating the contribution on $S \cap S^{n+1}$, we derive

$$\|(\mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n))_{S \setminus S^{n+1}}\|_2 \le \|(\mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n))_{S^{n+1} \setminus S}\|_2. \tag{3.12}$$

For the right-hand side, we have

$$\|(\mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n))_{S^{n+1} \setminus S}\|_2 = \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x})\big)_{S^{n+1} \setminus S}\|_2.$$

As for the left-hand side, we have

$$\begin{aligned}
\|(\mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n))_{S \setminus S^{n+1}}\|_2 &= \|(\mathbf{x} - \mathbf{x}^{n+1})_{\overline{S^{n+1}}} + \big((I - A^*A)(\mathbf{x}^n - \mathbf{x})\big)_{S \setminus S^{n+1}}\|_2 \\
&\ge \|(\mathbf{x} - \mathbf{x}^{n+1})_{\overline{S^{n+1}}}\|_2 - \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x})\big)_{S \setminus S^{n+1}}\|_2.
\end{aligned}$$

With $S \Delta S^{n+1}$ denoting the symmetric difference of the sets S and S^{n+1}, it follows that

$$\begin{aligned}
\|(\mathbf{x} - \mathbf{x}^{n+1})_{\overline{S^{n+1}}}\|_2 &\le \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x})\big)_{S \setminus S^{n+1}}\|_2 + \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x})\big)_{S^{n+1} \setminus S}\|_2 \\
&\le \sqrt{2}\, \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x})\big)_{S \Delta S^{n+1}}\|_2 \underset{(3.8)}{\le} \sqrt{2}\, \delta_{3s} \|\mathbf{x}^n - \mathbf{x}\|_2.
\end{aligned} \tag{3.13}$$

As a final step, we put (3.11) and (3.13) together to obtain

$$\|\mathbf{x}^{n+1} - \mathbf{x}\|_2 \leq \sqrt{\frac{2\delta_{3s}^2}{1-\delta_{2s}^2}}\, \|\mathbf{x}^n - \mathbf{x}\|_2.$$

The estimate (3.9) immediately follows. We point out that the multiplicative coefficient $\rho := \sqrt{2\delta_{3s}^2/(1-\delta_{2s}^2)}$ is less than one as soon as $2\delta_{3s}^2 < 1 - \delta_{2s}^2$. Since $\delta_{2s} \leq \delta_{3s}$, this occurs as soon as $\delta_{3s} < 1/\sqrt{3}$. □

As noticed earlier, the convergence requires a finite number of iterations, which can be estimated as follows.

COROLLARY 3.6. *Suppose that the matrix $A \in \mathbb{C}^{m\times N}$ satisfies $\delta_{3s} < 1/\sqrt{3}$. Then any s-sparse vector $\mathbf{x} \in \mathbb{C}^N$ is recovered by (HTP) with $\mathbf{y} = A\mathbf{x}$ in at most*

$$\left\lceil \frac{\ln\big(\sqrt{2/3}\,\|\mathbf{x}^0 - \mathbf{x}\|_2/\xi\big)}{\ln\big(1/\rho\big)} \right\rceil \quad \textit{iterations}, \tag{3.14}$$

where $\rho := \sqrt{2\delta_{3s}^2/(1-\delta_{2s}^2)}$ and ξ is the smallest nonzero entry of $\mathbf{x}$ in modulus.

Proof. We need to determine an integer n such that $S^n = S$, since then (HTP$_2$) implies $\mathbf{x}^n = \mathbf{x}$. According to the definition of S^n, this occurs if, for all $j \in S$ and all $\ell \in \overline{S}$, we have

$$|(\mathbf{x}^{n-1} + A^*A(\mathbf{x} - \mathbf{x}^{n-1}))_j| > |(\mathbf{x}^{n-1} + A^*A(\mathbf{x} - \mathbf{x}^{n-1}))_\ell|. \tag{3.15}$$

We observe that

$$|(\mathbf{x}^{n-1}+A^*A(\mathbf{x}-\mathbf{x}^{n-1}))_j| = |\mathbf{x}_j+\big((I-A^*A)(\mathbf{x}^{n-1}-\mathbf{x})\big)_j| \geq \xi-|\big((I-A^*A)(\mathbf{x}^{n-1}-\mathbf{x})\big)_j|,$$

and that

$$|(\mathbf{x}^{n-1} + A^*A(\mathbf{x} - \mathbf{x}^{n-1}))_\ell| = |\big((I - A^*A)(\mathbf{x}^{n-1} - \mathbf{x})\big)_\ell|.$$

Then, in view of

$$\begin{aligned} |\big((I - A^*A)(\mathbf{x}^{n-1} - \mathbf{x})\big)_j| &+ |\big((I - A^*A)(\mathbf{x}^{n-1} - \mathbf{x})\big)_\ell| \\ &\leq \sqrt{2}\,\|\big((I - A^*A)(\mathbf{x}^{n-1} - \mathbf{x})\big)_{\{j,\ell\}}\|_2 \leq \sqrt{2}\,\delta_{3s}\|\mathbf{x}^{n-1} - \mathbf{x}\|_2 \\ &= \sqrt{1-\delta_{2s}^2}\,\rho\,\|\mathbf{x}^{n-1} - \mathbf{x}\|_2 < \sqrt{2/3}\,\rho^n\|\mathbf{x}^0 - \mathbf{x}\|_2, \end{aligned}$$

we see that (3.15) is satisfied as soon as

$$\xi \geq \sqrt{2/3}\,\rho^n\|\mathbf{x}^0 - \mathbf{x}\|_2.$$

The smallest such integer n is the one given by (3.14). □

Turning our attention to the fast version of the Hard Thresholding Pursuit algorithm, it is interesting to notice that s-sparse recovery via (FHTP) is also guaranteed by the condition $\delta_{3s} < 1/\sqrt{3}$, independently on the number k of descent iterations used in (FHTP$_2$). Note that here we do not make the default choice for $t_{n+1,\ell}$ given by (2.1), but we simply choose $t_{n+1,\ell} = 1$, which in practice is not optimal. With $k = 0$, the result means that the classical IHT algorithm also allows s-sparse recovery as soon as $\delta_{3s} < 1/\sqrt{3}$, which incidentally improves the best condition [11] found in the current literature.

THEOREM 3.7. *Suppose that the $3s$th order restricted isometry constant of the measurement matrix $A \in \mathbb{C}^{m \times N}$ satisfies*

$$\delta_{3s} < \frac{1}{\sqrt{3}} \approx 0.57735.$$

Then, for any s-sparse $\mathbf{x} \in \mathbb{C}^N$, the sequence $(\mathbf{x}^n)$ defined by (FHTP) with $\mathbf{y} = A\mathbf{x}$, $k \geq 0$, and $t_{n+1,\ell} = 1$ converges towards $\mathbf{x}$ at a geometric rate given by

$$\|\mathbf{x}^n - \mathbf{x}\|_2 \leq \rho^n \, \|\mathbf{x}^0 - \mathbf{x}\|_2, \qquad \rho := \sqrt{\frac{\delta_{3s}^{2k+2}(1 - 3\delta_{3s}^2) + 2\delta_{3s}^2}{1 - \delta_{3s}^2}} < 1. \tag{3.16}$$

Proof. Exactly as in the proof of Theorem 3.5, we can still derive (3.13) from (FHTP$_1$), i.e.,

$$\|(\mathbf{x} - \mathbf{x}^{n+1})_{\overline{S^{n+1}}}\|_2^2 \leq 2\delta_{3s}^2 \|\mathbf{x} - \mathbf{x}^n\|_2^2. \tag{3.17}$$

Let us now examine the consequence of (FHTP$_2$). With $\mathbf{u}^{n+1,0} := \mathbf{x}^n$, we can write, for each $0 \leq \ell \leq k$,

$$\begin{aligned}
\|\mathbf{x} - \mathbf{u}^{n+1,\ell+1}\|_2^2 &= \|(\mathbf{x} - \mathbf{u}^{n+1,\ell+1})_{S^{n+1}}\|_2^2 + \|(\mathbf{x} - \mathbf{u}^{n+1,\ell+1})_{\overline{S^{n+1}}}\|_2^2 \\
&= \|\big((I - A^*A)(\mathbf{x} - \mathbf{u}^{n+1,\ell})\big)_{S^{n+1}}\|_2^2 + \|\mathbf{x}_{\overline{S^{n+1}}}\|_2^2 \\
&\leq \delta_{2s}^2 \|\mathbf{x} - \mathbf{u}^{n+1,\ell}\|_2^2 + \|\mathbf{x}_{\overline{S^{n+1}}}\|_2^2.
\end{aligned}$$

This yields, by immediate induction on ℓ,

$$\|\mathbf{x} - \mathbf{u}^{n+1,k+1}\|_2^2 \leq \delta_{2s}^{2k+2} \|\mathbf{x} - \mathbf{u}^{n+1,0}\|_2^2 + (\delta_{2s}^{2k} + \cdots + \delta_{2s}^2 + 1)\|\mathbf{x}_{\overline{S^{n+1}}}\|_2^2.$$

In other words, we have

$$\|\mathbf{x} - \mathbf{x}^{n+1}\|_2^2 \leq \delta_{2s}^{2k+2} \|\mathbf{x} - \mathbf{x}^n\|_2^2 + \frac{1 - \delta_{2s}^{2k+2}}{1 - \delta_{2s}^2} \|(\mathbf{x} - \mathbf{x}^{n+1})_{\overline{S^{n+1}}}\|_2^2. \tag{3.18}$$

From (3.17), (3.18), and the simple inequality $\delta_{2s} \leq \delta_{3s}$, we derive

$$\|\mathbf{x} - \mathbf{x}^{n+1}\|_2^2 \leq \frac{\delta_{3s}^{2k+2}(1 - 3\delta_{3s}^2) + 2\delta_{3s}^2}{1 - \delta_{3s}^2} \, \|\mathbf{x} - \mathbf{x}^n\|_2^2.$$

The estimate (3.16) immediately follows. We point out that, for any $k \geq 0$, the multiplicative coefficient ρ is less than one as soon as $\delta_{3s} < 1/\sqrt{3}$. □

Note that, although the sequence $(\mathbf{x}^n)$ does not converge towards $\mathbf{x}$ in a finite number of iterations, we can still estimate the number of iterations needed to approximate $\mathbf{x}$ with an ℓ_2-error not exceeding ϵ as

$$n_\epsilon = \left\lceil \frac{\ln\big(\|\mathbf{x}^0 - \mathbf{x}\|_2/\epsilon\big)}{\ln\big(1/\rho\big)} \right\rceil.$$

3.3. Approximate recovery of vectors from flawed measurements. In this section, we extend the previous results to the case of vectors that are not exactly sparse and that are not measured with perfect precision. Precisely, we prove that the HTP algorithm is stable and robust with respect to sparsity defect and to measurement error under the same sufficient condition on δ_{3s}. To this end, we need the observation that, for any $\mathbf{e} \in \mathbb{C}^m$,

$$\|(A^*\mathbf{e})_S\|_2 \le \sqrt{1+\delta_s}\,\|\mathbf{e}\|_2 \qquad \text{whenever } |S| \le s. \tag{3.19}$$

To see this, we write

$$\begin{aligned}\|(A^*\mathbf{e})_S\|_2^2 &= \langle A^*\mathbf{e}, (A^*\mathbf{e})_S\rangle = \langle \mathbf{e}, A\big((A^*\mathbf{e})_S\big)\rangle \le \|\mathbf{e}\|_2\,\|A\big((A^*\mathbf{e})_S\big)\|_2 \\ &\le \|\mathbf{e}\|_2\,\sqrt{1+\delta_s}\,\|(A^*\mathbf{e})_S\|_2,\end{aligned}$$

and we simplify by $\|(A^*\mathbf{e})_S\|_2$. Let us now state the main result of this subsection.

THEOREM 3.8. *Suppose that the $3s$th order restricted isometry constant of the measurement matrix $A \in \mathbb{C}^{m\times N}$ satisfies*

$$\delta_{3s} < \frac{1}{\sqrt{3}} \approx 0.57735.$$

Then, for any $\mathbf{x} \in \mathbb{C}^N$ and any $\mathbf{e} \in \mathbb{C}^m$, if S denotes an index set of s largest (in modulus) entries of $\mathbf{x}$, the sequence $(\mathbf{x}^n)$ defined by (HTP) with $\mathbf{y} = A\mathbf{x}+\mathbf{e}$ satisfies

$$\|\mathbf{x}^n - \mathbf{x}_S\|_2 \le \rho^n \|\mathbf{x}^0 - \mathbf{x}_S\|_2 + \tau\,\frac{1-\rho^n}{1-\rho}\,\|A\mathbf{x}_{\overline{S}} + \mathbf{e}\|_2, \qquad \textit{all } n \ge 0, \tag{3.20}$$

where

$$\rho := \sqrt{\frac{2\delta_{3s}^2}{1-\delta_{2s}^2}} < 1 \qquad \textit{and} \qquad \tau := \frac{\sqrt{2(1-\delta_{2s})} + \sqrt{1+\delta_s}}{1-\delta_{2s}} \le 5.15.$$

Proof. The proof follows the proof of Theorem 3.5 closely, starting with a consequence of (HTP$_2$) and continuing with a consequence of (HTP$_1$). We notice first that the orthogonality characterization (3.10) of $A\mathbf{x}^{n+1}$ is still valid, so that, writing $\mathbf{y} = A\mathbf{x}_S + \mathbf{e}'$ with $\mathbf{e}' := A\mathbf{x}_{\overline{S}} + \mathbf{e}$, we have

$$\langle \mathbf{x}^{n+1} - \mathbf{x}_S, A^*A\mathbf{z}\rangle = \langle \mathbf{e}', A\mathbf{z}\rangle \qquad \text{whenever } \mathrm{supp}(\mathbf{z}) \subseteq S^{n+1}.$$

We derive in particular

$$\begin{aligned}&\|(\mathbf{x}^{n+1} - \mathbf{x}_S)_{S^{n+1}}\|_2^2 = \langle \mathbf{x}^{n+1} - \mathbf{x}_S, (\mathbf{x}^{n+1} - \mathbf{x}_S)_{S^{n+1}}\rangle \\ &\quad = \langle \mathbf{x}^{n+1} - \mathbf{x}_S, (I - A^*A)\big((\mathbf{x}^{n+1} - \mathbf{x}_S)_{S^{n+1}}\big)\rangle + \langle \mathbf{e}', A\big((\mathbf{x}^{n+1} - \mathbf{x}_S)_{S^{n+1}}\big)\rangle \\ &\quad \underset{(3.7)-(3.19)}{\le} \delta_{2s}\|\mathbf{x}^{n+1} - \mathbf{x}_S\|_2\|(\mathbf{x}^{n+1} - \mathbf{x}_S)_{S^{n+1}}\|_2 + \|\mathbf{e}'\|_2\,\sqrt{1+\delta_s}\,\|(\mathbf{x}^{n+1} - \mathbf{x}_S)_{S^{n+1}}\|_2.\end{aligned}$$

After simplification, we have $\|(\mathbf{x}^{n+1} - \mathbf{x}_S)_{S^{n+1}}\|_2 \le \delta_{2s}\|\mathbf{x}^{n+1} - \mathbf{x}_S\|_2 + \sqrt{1+\delta_s}\,\|\mathbf{e}'\|_2$. It follows that

$$\begin{aligned}\|\mathbf{x}^{n+1} - \mathbf{x}_S\|_2^2 &= \|(\mathbf{x}^{n+1} - \mathbf{x}_S)_{\overline{S^{n+1}}}\|_2^2 + \|(\mathbf{x}^{n+1} - \mathbf{x}_S)_{S^{n+1}}\|_2^2 \\ &\le \|(\mathbf{x}^{n+1} - \mathbf{x}_S)_{\overline{S^{n+1}}}\|_2^2 + \big(\delta_{2s}\|\mathbf{x}^{n+1} - \mathbf{x}_S\|_2 + \sqrt{1+\delta_s}\,\|\mathbf{e}'\|_2\big)^2.\end{aligned}$$

This reads $P(\|\mathbf{x}^{n+1} - \mathbf{x}_S\|_2) \le 0$ for the quadratic polynomial defined by

$$P(t) := (1-\delta_{2s}^2)\,t^2 - (2\delta_{2s}\sqrt{1+\delta_s}\,\|\mathbf{e}'\|_2)\,t - (\|(\mathbf{x}^{n+1} - \mathbf{x}_S)_{\overline{S^{n+1}}}\|_2^2 + (1+\delta_s)\|\mathbf{e}'\|_2^2).$$

Hence $\|\mathbf{x}^{n+1} - \mathbf{x}_S\|_2$ is bounded by the largest root of P, i.e.,

$$\|\mathbf{x}^{n+1} - \mathbf{x}_S\|_2 \leq \frac{\delta_{2s}\sqrt{1+\delta_s}\,\|\mathbf{e}'\|_2 + \sqrt{(1-\delta_{2s}^2)\|(\mathbf{x}^{n+1}-\mathbf{x}_S)_{\overline{S^{n+1}}}\|_2^2 + (1+\delta_s)\|\mathbf{e}'\|_2^2}}{1-\delta_{2s}^2}.$$

Using the fact that $\sqrt{a^2+b^2} \leq a+b$ for $a, b \geq 0$, we obtain

$$\|\mathbf{x}^{n+1} - \mathbf{x}_S\|_2 \leq \frac{1}{\sqrt{1-\delta_{2s}^2}}\|(\mathbf{x}^{n+1}-\mathbf{x}_S)_{\overline{S^{n+1}}}\|_2 + \frac{\sqrt{1+\delta_s}}{1-\delta_{2s}}\|\mathbf{e}'\|_2. \tag{3.21}$$

We now notice that (HTP$_1$) still implies (3.12). For the right-hand side of (3.12), we have

$$\begin{aligned}\|(\mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n))_{S^{n+1}\setminus S}\|_2 &= \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x}_S) + A^*\mathbf{e}'\big)_{S^{n+1}\setminus S}\|_2\\ &\leq \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x}_S)\big)_{S^{n+1}\setminus S}\|_2 + \|(A^*\mathbf{e}')_{S^{n+1}\setminus S}\|_2.\end{aligned}$$

As for the left-hand side of (3.12), we have

$$\begin{aligned}\|(\mathbf{x}^n + A^*(\mathbf{y} - A\mathbf{x}^n))_{S\setminus S^{n+1}}\|_2 &= \|\big(\mathbf{x}_S + (I - A^*A)(\mathbf{x}^n - \mathbf{x}_S) + A^*\mathbf{e}'\big)_{S\setminus S^{n+1}}\|_2\\ \geq \|(\mathbf{x}_S - \mathbf{x}^{n+1})_{\overline{S^{n+1}}}\|_2 &- \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x}_S)\big)_{S\setminus S^{n+1}}\|_2 - \|(A^*\mathbf{e}')_{S\setminus S^{n+1}}\|_2.\end{aligned}$$

It follows that

$$\begin{aligned}\|(\mathbf{x}_S - \mathbf{x}^{n+1})_{\overline{S^{n+1}}}\|_2 &\leq \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x}_S)\big)_{S\setminus S^{n+1}}\|_2 + \|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x}_S)\big)_{S^{n+1}\setminus S}\|_2\\ &\quad + \|(A^*\mathbf{e}')_{S\setminus S^{n+1}}\|_2 + \|(A^*\mathbf{e}')_{S^{n+1}\setminus S}\|_2\\ &\leq \sqrt{2}\,\big[\|\big((I - A^*A)(\mathbf{x}^n - \mathbf{x}_S)\big)_{S\Delta S^{n+1}}\|_2 + \|(A^*\mathbf{e}')_{S\Delta S^{n+1}}\|_2\big]\\ &\underset{(3.8)-(3.19)}{\leq} \sqrt{2}\,\big[\delta_{3s}\|\mathbf{x}^n - \mathbf{x}_S\|_2 + \sqrt{1+\delta_{2s}}\,\|\mathbf{e}'\|_2\big].\end{aligned} \tag{3.22}$$

As a final step, we put (3.21) and (3.22) together to obtain

$$\|\mathbf{x}^{n+1} - \mathbf{x}_S\|_2 \leq \sqrt{\frac{2\delta_{3s}^2}{1-\delta_{2s}^2}}\,\|\mathbf{x}^n - \mathbf{x}_S\|_2 + \frac{\sqrt{2(1-\delta_{2s})} + \sqrt{1+\delta_s}}{1-\delta_{2s}}\,\|\mathbf{e}'\|_2.$$

Again, we point out that the multiplicative coefficient $\rho := \sqrt{2\delta_{3s}^2/(1-\delta_{2s}^2)}$ is less than one as soon as $\delta_{3s} < 1/\sqrt{3}$. In this case, the estimate (3.20) easily follows. □

We can deduce from Theorem 3.8 some error estimates that are comparable to the ones available for Basis Pursuit. We include the argument here because it does not seem to be standard, although somewhat known, see [18, Remark 2.3] and [1, p. 87]. It actually applies to any algorithm producing $(c\,s)$-sparse vectors for which the estimates (3.20) are available, hence also to IHT and to CoSaMP. A key inequality in the proof goes back to Stechkin, and reads, for $p \geq 1$,

$$\sigma_s(\mathbf{x})_p \leq \frac{1}{s^{1-1/p}}\|\mathbf{x}\|_1, \qquad \mathbf{x} \in \mathbb{C}^N. \tag{3.23}$$

An improved inequality can be found in [13] for $p = 2$, and a sharp and more general inequality can be found in [11] for any $p \geq 1$.

COROLLARY 3.9. *Suppose that the matrix $A \in \mathbb{C}^{m \times N}$ satisfies $\delta_{6s} < 1/\sqrt{3}$. Then, for any $\mathbf{x} \in \mathbb{C}^N$ and any $\mathbf{e} \in \mathbb{C}^m$, every cluster point $\mathbf{x}^\star$ of the sequence $(\mathbf{x}^n)$ defined by (HTP) with s replaced by $2s$ and with $\mathbf{y} = A\mathbf{x} + \mathbf{e}$ satisfies*

$$\|\mathbf{x} - \mathbf{x}^\star\|_p \le \frac{C}{s^{1-1/p}} \sigma_s(\mathbf{x})_1 + D s^{1/p-1/2} \|\mathbf{e}\|_2, \qquad 1 \le p \le 2,$$

where the constants C and D depend only on δ_{6s}.

Proof. Let S_0 be an index set of s largest components of $\mathbf{x}$, S_1 an index set of s next largest components of $\mathbf{x}$, etc. It is classical to notice that, for $k \ge 1$,

$$\|\mathbf{x}_{S_k}\|_2 \le \frac{\|\mathbf{x}_{S_{k-1}}\|_1}{s^{1/2}}. \tag{3.24}$$

We then obtain

$$\begin{aligned}
\|\mathbf{x} - \mathbf{x}^\star\|_p &\le \|\mathbf{x}_{\overline{S_0 \cup S_1}}\|_p + \|\mathbf{x}^\star - \mathbf{x}_{S_0 \cup S_1}\|_p \le \|\mathbf{x}_{\overline{S_0 \cup S_1}}\|_p + (4s)^{1/p-1/2} \|\mathbf{x}^\star - \mathbf{x}_{S_0 \cup S_1}\|_2 \\
&\underset{(3.20)}{\le} \|\mathbf{x}_{\overline{S_0 \cup S_1}}\|_p + (4s)^{1/p-1/2} \frac{\tau}{1-\rho} \|A\mathbf{x}_{\overline{S_0 \cup S_1}} + \mathbf{e}\|_2 \\
&\underset{(3.23)}{\le} \frac{1}{s^{1-1/p}} \|\mathbf{x}_{\overline{S_0}}\|_1 + (4s)^{1/p-1/2} \frac{\tau}{1-\rho} \|A\mathbf{x}_{\overline{S_0 \cup S_1}} + \mathbf{e}\|_2 \\
&\le \frac{1}{s^{1-1/p}} \sigma_s(\mathbf{x})_1 + (4s)^{1/p-1/2} \frac{\tau}{1-\rho} \Big(\|A\mathbf{x}_{S_2}\|_2 + \|A\mathbf{x}_{S_3}\|_2 + \cdots + \|\mathbf{e}\|_2 \Big) \\
&\le \frac{1}{s^{1-1/p}} \sigma_s(\mathbf{x})_1 + (4s)^{1/p-1/2} \frac{\tau}{1-\rho} \Big(\sqrt{1+\delta_s} \, (\|\mathbf{x}_{S_2}\|_2 + \|\mathbf{x}_{S_3}\|_2 + \cdots) + \|\mathbf{e}\|_2 \Big) \\
&\underset{(3.24)}{\le} \frac{1}{s^{1-1/p}} \sigma_s(\mathbf{x})_1 + (4s)^{1/p-1/2} \frac{\tau}{1-\rho} \Big(\sqrt{1+\delta_s} \, \frac{\|\mathbf{x}_{\overline{S_0}}\|_1}{s^{1/2}} + \|\mathbf{e}\|_2 \Big) \\
&= \frac{C}{s^{1-1/p}} \sigma_s(\mathbf{x})_1 + D s^{1/p-1/2} \|\mathbf{e}\|_2,
\end{aligned}$$

where $C \le 1 + 4^{1/p-1/2} \tau \sqrt{1+\delta_{6s}}/(1-\rho)$ and $D := 4^{1/p-1/2} \tau/(1-\rho)$. □

Remark: Theorem 3.8 does not guarantee the observed convergence of the Hard Thresholding Pursuit algorithm. In fact, for a small restricted isometry constant, say $\delta_{3s} \le 1/2$, Proposition 3.2 does not explain it either, except in the restrictive case $m \ge cN$ for some absolute constant $c > 0$. Indeed, if $\|A\|_{2 \to 2} < 1$, for any cluster point $\mathbf{x}^\star$ of the sequence $(\mathbf{x}^n)$ defined by (HTP) with $\mathbf{y} = A\mathbf{x} + \mathbf{e}$, we would derive from (3.20) that

$$\|\mathbf{x} - \mathbf{x}^\star\|_2 \le C \, \sigma_s(\mathbf{x})_2 + D \|\mathbf{e}\|_2, \qquad C = 1 + D\|A\|_{2 \to 2} < 1 + D,$$

for some absolute constant $D > 0$. But the ℓ_2-instance optimal estimate obtained by setting $\mathbf{e} = 0$ is known to hold only when $m \ge cN$ with c depending only on C, see [7]. This justifies our statement. However, if $\delta_{3s} \le 0.4058$, one can guarantee both convergence and estimates of type (3.20) for HTP$^\mu$ with $\mu \approx 0.7113$. Indeed, we notice that applying HTP$^\mu$ with inputs $\mathbf{y} \in \mathbb{C}^m$ and $A \in \mathbb{C}^{m \times N}$ is the same as applying HTP with inputs $\mathbf{y}' := \sqrt{\mu}\, \mathbf{y} \in \mathbb{C}^m$ and $A' := \sqrt{\mu}\, A \in \mathbb{C}^{m \times N}$. Therefore, our double objective will be met as soon as $\mu(1 + \delta_{2s}(A)) < 1$ and $\delta_{3s}(A') < 1/\sqrt{3}$. With $\delta_{3s} := \delta_{3s}(A)$, the former is implied by

$$\mu < \frac{1}{1 + \delta_{3s}}, \tag{3.25}$$

while the latter, in view of $\delta_{3s}(A') \leq \max\{1-\mu(1-\delta_{3s}), \mu(1+\delta_{3s})-1\}$, is implied by

$$\frac{1-1/\sqrt{3}}{1-\delta_{3s}} < \mu < \frac{1+1/\sqrt{3}}{1+\delta_{3s}}. \tag{3.26}$$

These two conditions can be fulfilled simultaneously as soon as the upper bound in (3.25) is larger than the lower bound in (3.26) by choosing μ between these two values, i.e., as soon as

$$\delta_{3s} < \frac{1}{2\sqrt{3}-1} \approx 0.4058 \qquad \text{by choosing} \quad \mu = 1 - \frac{1}{2\sqrt{3}} \approx 0.7113.$$

4. Computational investigations.

Even though we have obtained better theoretical guarantees for the Hard Thresholding Pursuit algorithms than for other algorithms, this does not say much about empirical performances, because there is no way to verify conditions of the type $\delta_t \leq \delta_*$. This section aims at supporting the claim that the proposed algorithms are also competitive in practice — at least in the situations considered. Of course, one has to be cautious about the validity of the conclusions drawn from our computational investigations: our numerical experiments only involved the dimensions $N = 1000$ and $m = 200$, they assess average situations while theoretical considerations assess worst-case situation, etc. The interested readers are invited to make up their own opinion based on the codes for HTP and FHTP available on the author's web page. To allow for equitable comparison, the codes for the algorithms tested here — except ℓ_1-magic and NESTA — have been implemented by the author on the same model as the codes for HTP and FHTP. These implementations — and perhaps the usage of ℓ_1-magic and NESTA — may not be optimal.

4.1. Number of iterations and CSMP algorithms.

The first issue to be investigated concerns the number of iterations required by the HTP algorithm in comparison with algorithms in the CSMP family. We recall that, assuming convergence of the HTP algorithm, this convergence only requires a finite number of iterations. The same holds for the CoSaMP and SP algorithms, since the sequences (U^n) and (x^n) they produce are also eventually periodic. The stopping criterion $U^{n+1} = U^n$ is natural for these algorithms, too, and was therefore incorporated in our implementations. Although acceptable outputs may be obtained earlier (with $\mathrm{e} = 0$, note in particular that $\mathrm{x}^n \approx \mathrm{x}$ yields $A^*(\mathrm{y} - A\mathrm{x}^n) \approx 0$, whose s or $2s$ largest entries are likely to change with every n), our tests did not reveal any noticeable difference with the stopping criterion $\|\mathrm{y} - A\mathrm{x}^n\|_2 < 10^{-4}\,\|\mathrm{y}\|_2$. The experiment presented here consisted in running the HTP, SP, and CoSaMP algorithms 500 times — 100 realizations of Gaussian matrices A and 5 realizations of s-sparse Gaussian vectors x per matrix realization — with common inputs s, A, and $\mathrm{y} = A\mathrm{x}$. For each $1 \leq s \leq 120$, the number of successful reconstructions for these algorithms (and others) is recorded in the first plot of Figure 4.4. This shows that, in the purely Gaussian setting, the HTP algorithm performs slightly better than the CSMP algorithms. The most significant advantage appears to be the fewer number of iterations required for convergence. This is reported in Figure 4.1, where we discriminated between successful reconstructions (mostly occurring for $s \leq 60$) in the left column and unsuccessful reconstructions (mostly occurring for $s \geq 60$) in the right column. The top row of plots shows the number of iterations averaged over all trials, while the bottom row shows the largest

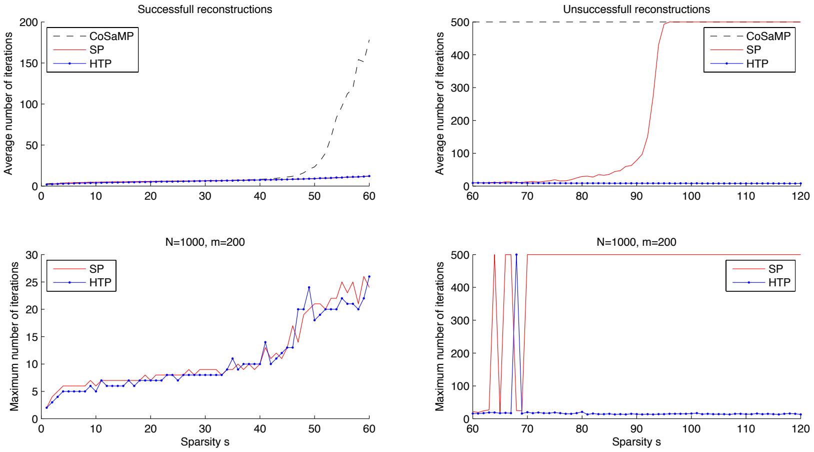

FIG. 4.1. *Number of iterations for CSMP and HTP algorithms (Gaussian matrices and vectors)*

number of iterations encountered. In the successful case, we observe that the CoSaMP algorithm requires more iterations than the SP and HTP algorithms, which are comparable in this respect (we keep in mind, however, that one SP iteration involves twice as many orthogonal projections as one HTP iteration). In the unsuccessful case, the advantage of the HTP algorithm is even more apparent, as CSMP algorithms fail to converge (the number of iterations reaches the imposed limit of 500 iterations). For the HTP algorithm, we point out that cases of nonconvergence were extremely rare. This raises the question of finding suitable conditions to guarantee its convergence (see the remark at the end of Subsection 3.3 in this respect).

4.2. Choices of parameters and thresholding algorithms. The second issue to be investigated concerns the performances of different algorithms from the IHT and HTP families. Precisely, the algorithms of interest here are the classical Iterative Hard Thresholding, the Iterative Hard Thresholding with parameter $\mu = 1/3$ (as advocated in [12]), the Normalized Iterative Hard Thresholding of [4], the Normalized Hard Thresholding Pursuit, the Hard Thresholding Pursuit with default parameter $\mu = 1$ as well as with parameters $\mu = 0.71$ (see remark at the end of Section 3.3) and $\mu = 1.6$, and the Fast Hard Thresholding Pursuit with various parameters μ, k, and $t_{n+1,\ell}$. For all algorithms — other than HTP, HTP$^{0.71}$, and HTP$^{1.6}$ — the stopping criterion $\|\mathbf{y} - A\mathbf{x}^n\|_2 < 10^{-5}\|\mathbf{y}\|_2$ was used, and a maximum of 1000 iterations was imposed. All the algorithms were run 500 times — 100 realizations of Gaussian matrices A and 5 realizations of s-sparse Gaussian vectors $\mathbf{x}$ per matrix realization — with common inputs s, A, and $\mathbf{y} = A\mathbf{x}$. A successful reconstruction was recorded if $\|\mathbf{x} - \mathbf{x}^n\|_2 < 10^{-4}\|\mathbf{x}\|_2$. For each $1 \leq s \leq 120$, the number of successful reconstructions is reported in Figure 4.2. The latter shows in particular that HTP and FHTP perform slightly better than NIHT, that the normalized versions of HTP and FHTP do not improve performance, and that the choice $t_{n+1,\ell} = 1$ degrades the performance of FHTP with the default choice $t_{n+1,\ell}$ given by (2.1). This latter observation echoes the drastic improvement generated by step (HTP$_2$) when added in the classical IHT algorithm. We also notice from Figure 4.2 that the performance of HTP$^\mu$ increases with μ in a neighborhood of one in this purely Gaussian setting, but the situation changes in other settings, see Figure 4.4. We now focus on the reconstruction

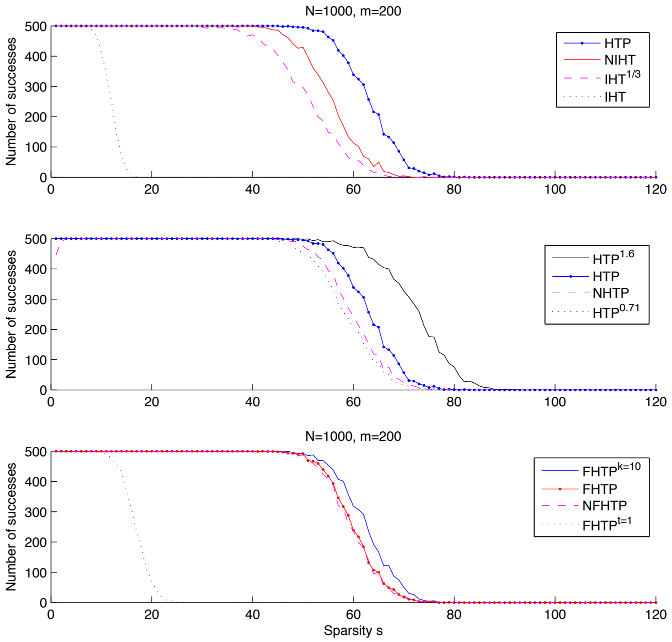

FIG. 4.2. *Number of successes for IHT and HTP algorithms (Gaussian matrices and vectors)*

time for the algorithms considered here, discriminating again between successful and unsuccessful reconstructions. As the top row of Figure 4.3 suggests, the HTP algorithm is faster than the NIHT, IHT$^{1/3}$, and classical IHT algorithms in both successful and unsuccessful cases. This is essentially due to the fewer iterations required by HTP. Indeed, the average-time-per-reconstruction pattern follows very closely the average-number-of-iterations pattern (not shown). For larger scales, it is unclear if the lower costs per iteration of the NIHT, IHT$^{1/3}$, and IHT algorithms would compensate the higher number of iterations they require. For the current scale, these algorithms do not seem to converge in the unsuccessful case, as they reach the maximum number of iterations allowed. The middle row of Figure 4.3 then suggests that the HTP$^{1.6}$, HTP, NHTP, and HTP$^{0.71}$ algorithms are roughly similar in terms of speed, with the exception of HTP$^{1.6}$ which is much slower in the unsuccessful case (hence HTP$^{1.3}$ was displayed instead). Again, this is because the average-time-per-reconstruction pattern follows very closely the average-number-of-iterations pattern (not shown). As for the bottom row of Figure 4.3, it suggests that in the successful case the normalized version of FHTP with $k = 3$ descent steps is actually slower than FTHP with default parameters $k = 3$, $t_{n+1,\ell}$ given by (2.1), with parameters $k = 10$, $t_{n+1,\ell}$ given by (2.1), and with parameters $k = 3$, $t_{n+1,\ell} = 1$, which are all comparable in terms of speed. In the unsuccessful case, FHTP with parameters $k = 3$, $t_{n+1,\ell} = 1$ becomes the fastest of these algorithms, because it does not reach the allowed maximum number of iterations (figure not shown).

4.3. Non-Gaussian settings and other classical algorithms. The final issue to be investigated concerns the relative performance of the HTP algorithm compared with other classical algorithms, namely the NIHT, CoSaMP, SP, and BP algorithms. This experiment is more exhaustive than the previous ones since

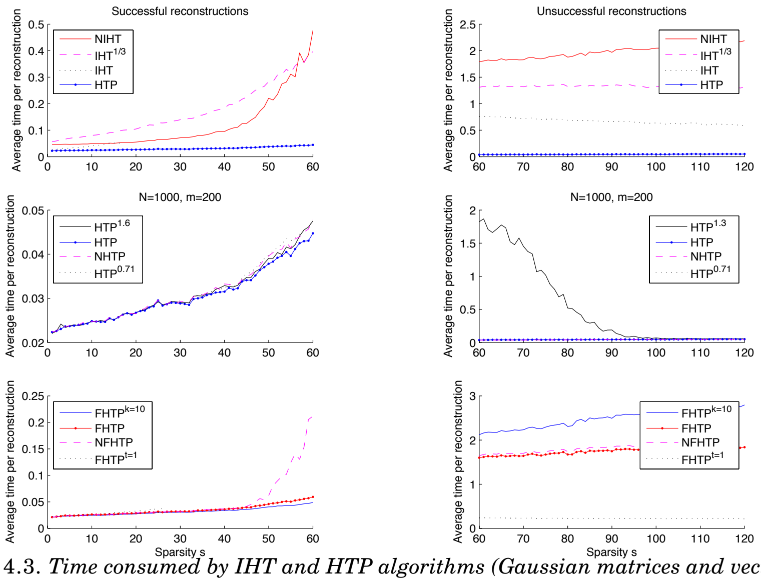

FIG. 4.3. *Time consumed by IHT and HTP algorithms (Gaussian matrices and vectors)*

not only Gaussian matrices and sparse Gaussian vectors are tested, but also Bernoulli and partial Fourier matrices and sparse Bernoulli vectors. The stopping criterion chosen for NIHT, CoSaMP, and SP was $\|\mathbf{y} - A\mathbf{x}^n\|_2 < 10^{-5}\|\mathbf{y}\|_2$, and a maximum of 500 iterations was imposed. All the algorithms were run 500 times — 100 realizations of Gaussian matrices A and 5 realizations of s-sparse Gaussian vectors x per matrix realization — with common inputs s, A, and $\mathbf{y} = A\mathrm{x}$ (it is worth recalling that BP does not actually need s as an input). A successful reconstruction was recorded if $\|\mathbf{x} - \mathbf{x}^n\|_2 < 10^{-4}\|\mathbf{x}\|_2$. For each $1 \leq s \leq 120$, the number of successful reconstructions is reported in Figure 4.4, where the top row of plots corresponds to Gaussian matrices, the middle row to Bernoulli matrices, and the third row to partial Fourier matrices, while the first column of plots corresponds to sparse Gaussian vectors and the second column to sparse Bernoulli vectors. We observe that the HTP algorithms, especially HTP$^{1.6}$, outperform other algorithms for Gaussian vectors, but not for Bernoulli vectors. Such a phenomenon was already observed in [8] when comparing the SP and BP algorithms. Note that the BP algorithm behaves similarly for Gaussian or Bernoulli vectors, which is consistent with the theoretical observation that the recovery of a sparse vector via ℓ_1-minimization depends (in the real setting) only on the sign pattern of this vector. The time consumed by each algorithm to perform the experiment is reported in the legends of Figure 4.4, where the first number corresponds to the range $1 \leq s \leq 60$ and the second number to the range $61 \leq s \leq 120$. Especially in the latter range, HTP appears to be the fastest of the algorithms considered here.

5. Conclusion. We have introduced a new iterative algorithm, called Hard Thresholding Pursuit, designed to find sparse solutions of underdetermined linear systems. One of the strong features of the algorithm is its speed, which is accounted for by its simplicity and by the fewer number of iterations needed for its convergence. We have also given an elegant proof of its good theoretical performance, as we have shown that the Hard Thresholding Pursuit algorithm finds any s-sparse solution of a linear system (in a stable and robust way) if the re-

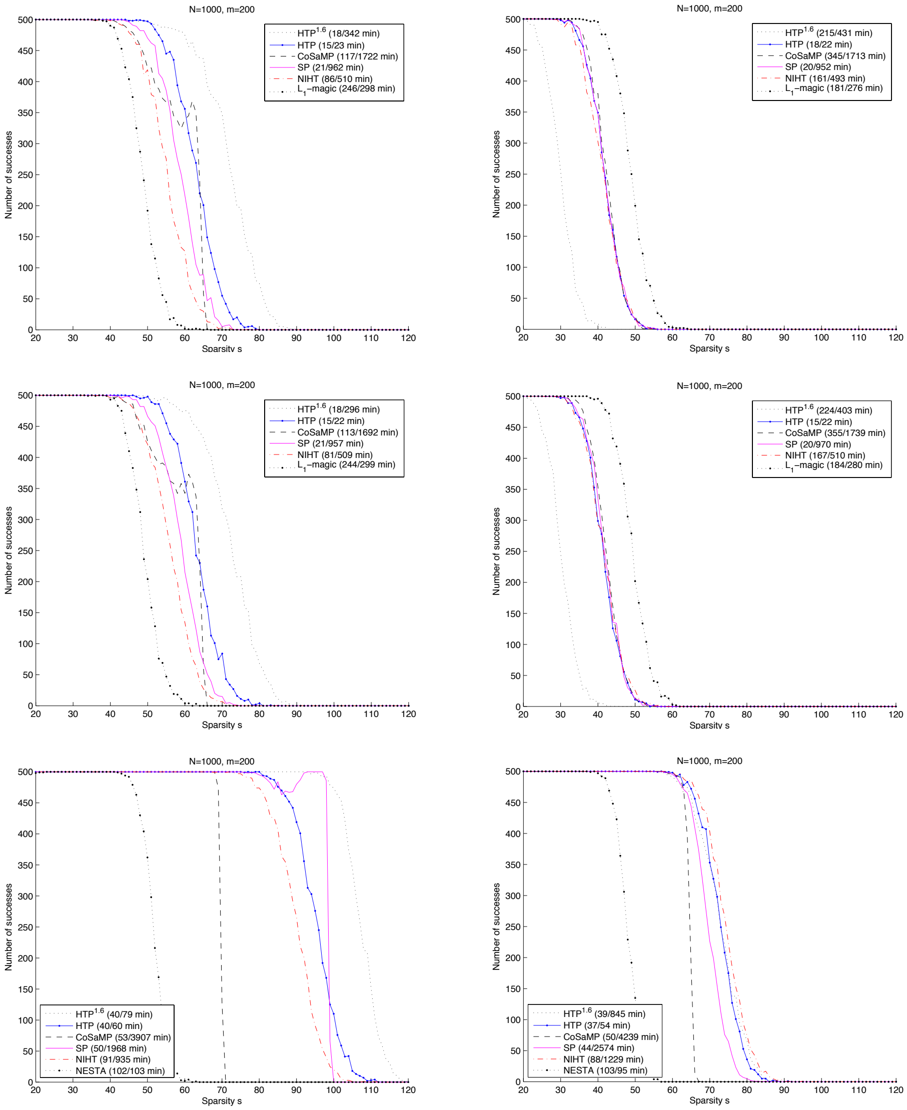

FIG. 4.4. *Comparison of different algorithms (top: Gauss, middle: Bernoulli, bottom: Fourier) with sparse Gaussian (left) and Bernoulli (right) vectors*

stricted isometry constant of the matrix of the system satisfies $\delta_{3s} < 1/\sqrt{3}$. We have finally conducted some numerical experiments with random linear systems to illustrate the fine empirical performance of the algorithm compared to other classical algorithms. Experiments on realistic situations are now needed to truly validate the practical performance of the Hard Thresholding Pursuit algorithm.

Acknowledgments. The author wishes to thank Albert Cohen and Rémi Gribonval for a discussion on an early version of this manuscript, and more generally for the opportunity to work on the project ECHANGE. Some preliminary results of this article were presented during a workshop in Oberwolfach, and the author wishes to thank Rachel Ward for calling the preprint [14] to his attention. This preprint considered an algorithm similar to HTP$^{\mu}$ with $\mu = 1/(1 + \delta_{2s})$, which was called Hybrid Iterative Thresholding. Despite the drawback that a prior knowledge of δ_{2s} is needed, theoretical results analog to Theorems 3.5 and 3.8 were established under the more demanding conditions $\delta_{2s} < 1/3$ and $\delta_{2s} < \sqrt{5} - 2$, respectively. The author also wishes to thank Ambuj Tewari for calling the articles [15] and [16] to his attention. They both considered the Hard Thresholding Pursuit algorithm for exactly sparse vectors measured without error. In [16], the analog of HTP was called Iterative Thresholding algorithm with Inversion. Its convergence to the targeted s-sparse vector in only s iterations was established under the condition $3\mu s < 1$, where μ is the coherence of the measurement matrix. In [15], the analog of HTP$^{\mu}$ was called Singular Value Projection-Newton (because it applied to matrix problems, too). With $\mu = 4/3$, its convergence to the targeted s-sparse vector in a finite number of iterations was established under the condition $\delta_{2s} < 1/3$. Finally, the author would like to thank Charles Soussen for pointing out his initial misreading of the Subspace Pursuit algorithm.

REFERENCES

[1] R. BERINDE, *Advances in sparse signal recovery methods*, Masters thesis, 2009.

[2] T. BLUMENSATH AND M.E. DAVIES, *Iterative Thresholding for Sparse Approximations*, J. Fourier Anal. Appl., 14 (2008), pp.629–654.

[3] T. BLUMENSATH AND M.E. DAVIES, *Iterative hard thresholding for compressed sensing*, Appl. Comput. Harmon. Anal., 27 (2009), pp.265–274.

[4] T. BLUMENSATH AND M.E. DAVIES, *Normalized iterative hard thresholding: guaranteed stability and performance*, IEEE Journal of selected topics in signal processing, 4 (2010), pp.298–309.

[5] T.T. CAI, L. WANG, AND G. XU, *Shifting inequality and recovery of sparse signals*, IEEE Transactions on Signal Processing, 58 (2010), pp.1300–1308.

[6] E. CANDÈS AND T. TAO, *Decoding by linear programing*, IEEE Trans. Inf. Theory, 51 (2005), pp.4203–4215.

[7] A. COHEN, W. DAHMEN, AND R.A. DEVORE, *Compressed sensing and best k-term approximation*, J. Amer. Math. Soc., 22 (2009), pp.211–231.

[8] W. DAI AND O. MILENKOVIC, *Subspace pursuit for compressive sensing signal reconstruction*, IEEE Trans. Inform. Theory, 55 (2009), pp.2230–2249.

[9] D. DONOHO AND J. TANNER, *Precise undersampling theorems*, Proceedings of the IEEE, 98 (2010), pp.913–924.

[10] S. FOUCART, *A note on guaranteed sparse recovery via ℓ_1-minimization*, Applied and Comput. Harmonic Analysis, 29 (2010), pp.97–103.

[11] S. FOUCART, *Sparse Recovery Algorithms: Sufficient conditions in terms of restricted isometry constants*, Proceedings of the 13th International Conference on Approximation Theory, Accepted.

[12] R. GARG AND R. KHANDEKAR, *Gradient descent with sparsification: An iterative algorithm for sparse recovery with restricted isometry property*, in Proceedings Of The 26th International Conference on Machine Learning, L. Bottou and M. Littman, eds., pp.337–344.

[13] A. GILBERT, M. STRAUSS, J. TROPP, AND R. VERSHYNIN, *One sketch for all: Fast algorithms for compressed sensing*, in Proceedings Of The 39th ACM Symp. Theory Of Computing, San Diego, 2007.

[14] S. HUANG AND J. ZHU, *Hybrid iterative thresholding for sparse signal recovery from incomplete measurements*, Preprint.

[15] P. JAIN, R. MEKA, AND I. S. DHILLON, *Guaranteed Rank Minimization via Singular Value Projection: Supplementary Material*, NIPS 2010.

[16] A. MALEKI, *Convergence analysis of iterative thresholding algorithms*, Proc. of Allerton Conference on Communication, Control, and Computing, 2009.

[17] S.G MALLAT AND Z. ZHANG, *Matching pursuits with time-frequency dictionaries*, IEEE Trans. Signal Process., 41 (1993), pp.3397–3415.

[18] D. NEEDELL AND J.A. TROPP, *CoSaMP: Iterative signal recovery from incomplete and inaccurate samples*, Appl. Comput. Harmon. Anal., 26 (2009), pp.301–32.

Appendix. Here is a justification of the fact that $\delta_t < \delta_*$ implies $m \geq c\,t/\delta_*^2$.

THEOREM 5.1. *For $A \in \mathbb{C}^{m \times N}$ and $2 \leq t \leq N$, one has*

$$m \geq c\,\frac{t}{\delta_t^2},$$

provided $N \geq C\,m$ and $\delta_t \leq \delta_*$, with e.g. $c = 1/162$, $C = 30$, and $\delta_* = 2/3$.

Proof. Let us first point out that the previous statement cannot hold for $t = 1$, as $\delta_1 = 0$ if all the columns of A have ℓ_2-norm equal to 1. In this case, we also note that the constant δ_2 reduces to the coherence μ of $A \in \mathbb{C}^{m \times N}$, and, specifying $t = 2$ in the argument below, we recognize a proof of the Welch bound $\mu \geq \sqrt{(N-m)/(m(N-1))}$. Let us now set $s := \lfloor t/2 \rfloor \geq 1$, and let us decompose the matrix A in blocks of size $m \times s$ — except possibly the last one — as

$$A = \left[\begin{array}{c|c|c|c} A_1 & A_2 & \cdots & A_n \end{array}\right], \qquad N \leq ns.$$

We recall that, for all $1 \leq i \neq j \leq n$,

$$\|A_i^* A_i - I_s\|_{2\to2} \leq \delta_s \leq \delta_t, \qquad \|A_i^* A_j\|_{2\to2} \leq \delta_{2s} \leq \delta_t,$$

so that the eigenvalues of $A_i^* A_i$ and the singular values of $A_i^* A_j$ satisfy

$$1 - \delta_t \leq \lambda_k(A_i^* A_i) \leq 1 + \delta_t, \qquad \sigma_k(A_i^* A_j) \leq \delta_t.$$

Let us introduce the matrices

$$H := A\,A^* \in \mathbb{C}^{m \times m}, \qquad G := A^* A = \left[A_i^* A_j\right]_{1 \leq i,j \leq n} \in \mathbb{C}^{N \times N}.$$

On the one hand, we have the lower bound

$$\mathrm{tr}(H) = \mathrm{tr}(G) = \sum_{i=1}^{n} \mathrm{tr}(A_i^* A_i) = \sum_{i=1}^{n} \sum_{k=1}^{s} \lambda_k(A_i^* A_i) \geq n\,s\,(1 - \delta_t). \tag{5.1}$$

On the other hand, writing $\langle M, N\rangle_F := \mathrm{tr}(N^* M)$ for the Frobenius inner product, we have

$$\mathrm{tr}(H)^2 = \langle I_m, H\rangle_F^2 \leq \|I_m\|_F^2 \|H\|_F^2 = m\,\mathrm{tr}(H^* H).$$

Then, in view of

$$\mathrm{tr}(H^* H) = \mathrm{tr}(A\,A^* A\,A^*) = \mathrm{tr}(A^* A\,A^* A) = \mathrm{tr}(G\,G^*) = \sum_{i=1}^{n} \mathrm{tr}\Big(\sum_{j=1}^{m} A_i^* A_j A_j^* A_i\Big)$$

$$= \sum_{1 \leq i \neq j \leq n} \sum_{k=1}^{s} \sigma_s(A_i^* A_j)^2 + \sum_{1 \leq i \leq n} \sum_{k=1}^{s} \lambda_k(A_i^* A_i)^2 \leq n\,(n-1)\,s\,\delta_t^2 + n\,s\,(1 + \delta_t)^2,$$

we derive the upper bound

$$\mathrm{tr}(H)^2 \leq m\,n\,s\,\big((n-1)\,\delta_t^2 + (1 + \delta_t)^2\big). \tag{5.2}$$

Combining the bounds (5.1) and (5.2) yields

$$m \geq \frac{n\,s\,(1 - \delta_t)^2}{(n-1)\,\delta_t^2 + (1 + \delta_t)^2}.$$

If $(n-1)\,\delta_t^2 < (1+\delta_t)^2/5$, we would obtain, using $\delta_t \le 2/3$,

$$m > \frac{n\,s\,(1-\delta_t)^2}{6(1+\delta_t)^2/5} \ge \frac{5(1-\delta_t)^2}{6(1+\delta_t)^2}\,N \ge \frac{1}{30}\,N,$$

which contradicts our assumption. We therefore have $(n-1)\,\delta_t^2 \ge (1+\delta_t)^2/5$, which yields, using $\delta_t \le 2/3$ again and $t \le 3s$,

$$m \ge \frac{n\,s\,(1-\delta_t)^2}{6(n-1)\,\delta_t^2} \ge \frac{1}{54}\frac{s}{\delta_t^2} \ge \frac{1}{162}\frac{t}{\delta_t^2}.$$

This is the announced result. □