SIAM REVIEW
Vol. 51, No. 1, pp. 34–81

# From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images*

Alfred M. Bruckstein[†]
David L. Donoho[‡]
Michael Elad[§]

**Abstract.** A full-rank matrix $\mathbf{A} \in \mathbb{R}^{n\times m}$ with $n < m$ generates an underdetermined system of linear equations $\mathbf{Ax} = \mathbf{b}$ having infinitely many solutions. Suppose we seek the sparsest solution, i.e., the one with the fewest nonzero entries. Can it ever be unique? If so, when? As optimization of sparsity is combinatorial in nature, are there efficient methods for finding the sparsest solution? These questions have been answered positively and constructively in recent years, exposing a wide variety of surprising phenomena, in particular the existence of easily verifiable conditions under which optimally sparse solutions can be found by concrete, effective computational methods. Such theoretical results inspire a bold perspective on some important practical problems in signal and image processing. Several well-known signal and image processing problems can be cast as demanding solutions of undetermined systems of equations. Such problems have previously seemed, to many, intractable, but there is considerable evidence that these problems often have sparse solutions. Hence, advances in finding sparse solutions to underdetermined systems have energized research on such signal and image processing problems—to striking effect. In this paper we review the theoretical results on sparse solutions of linear systems, empirical results on sparse modeling of signals and images, and recent applications in inverse problems and compression in image processing. This work lies at the intersection of signal processing and applied mathematics, and arose initially from the wavelets and harmonic analysis research communities. The aim of this paper is to introduce a few key notions and applications connected to sparsity, targeting newcomers interested in either the mathematical aspects of this area or its applications.

**Key words.** underdetermined, linear system of equations, redundant, overcomplete, sparse representation, sparse coding, basis pursuit, matching pursuit, mutual coherence, Sparse-Land, dictionary learning, inverse problems, denoising, compression

**AMS subject classifications.** 68U10, 94A08, 15A29, 15A06, 90C25

**DOI.** 10.1137/060657704

## 1. Introduction.

A central achievement of classical linear algebra was a thorough examination of the problem of solving systems of linear equations. The results—

[*] Received by the editors April 20, 2006; accepted for publication (in revised form) July 23, 2007; published electronically February 5, 2009. This work was supported in part by the Israel Science Foundation (ISF) grant 796/05, by the Binational Science Foundation (BSF) grant 2004199, and by an Applied Materials research fund.

http://www.siam.org/journals/sirev/51-1/65770.html

[†] Computer Science Department, The Technion–Israel Institute of Technology, Haifa, 32000 Israel (freddy@cs.technion.ac.il).

[‡] Statistics Department, Stanford University, Stanford, CA 94305 (donoho@stat.stanford.edu).

[§] Corresponding author: Computer Science Department, The Technion–Israel Institute of Technology, Haifa, 32000 Israel (elad@cs.technion.ac.il).

definite, timeless, and profound—give the subject a completely settled appearance. Surprisingly, within this well-understood arena there is an elementary problem which only recently has been explored in depth; we will see that this problem has surprising answers and inspires numerous practical developments.

**1.1. Sparse Solutions of Linear Systems of Equations?** Consider a full-rank matrix $\mathbf{A} \in \mathbb{R}^{n\times m}$ with $n < m$, and define the underdetermined linear system of equations $\mathbf{Ax} = \mathbf{b}$. This system has infinitely many solutions; if one desires to narrow the choice to one well-defined solution, additional criteria are needed. Here is a familiar way to do this. Introduce a function $J(\mathbf{x})$ to evaluate the desirability of a would-be solution $\mathbf{x}$, with smaller values being preferred. Define the general optimization problem $(P_J)$,

$$(1) \qquad (P_J): \qquad \min_{\mathbf{x}} \ J(\mathbf{x}) \ \text{ subject to } \ \mathbf{b} = \mathbf{Ax}.$$

Selecting a strictly convex function $J(\cdot)$ guarantees a unique solution. Most readers are familiar with this approach from the case where $J(\mathbf{x})$ is the squared Euclidean norm $\|\mathbf{x}\|_2^2$. The problem $(P_2)$ (say), which results from that choice, has in fact a unique solution $\hat{\mathbf{x}}$—the so-called minimum-norm solution; this is given explicitly by

$$\hat{\mathbf{x}}_2 = \mathbf{A}^+\mathbf{b} = \mathbf{A}^T(\mathbf{AA}^T)^{-1}\mathbf{b}.$$

The squared $\ell_2$ norm is of course a measure of energy; in this paper, we consider instead measures of *sparsity*. A very simple and intuitive measure of sparsity of a vector $\mathbf{x}$ simply involves the number of nonzero entries in $\mathbf{x}$; the vector is sparse if there are few nonzeros among the possible entries in $\mathbf{x}$. It will be convenient to introduce the $\ell_0$ "norm"

$$\|\mathbf{x}\|_0 = \#\{i : x_i \neq 0\}.$$

Thus if $\|\mathbf{x}\|_0 \ll m$, $\mathbf{x}$ is sparse.

Consider the problem $(P_0)$ obtained from the general prescription $(P_J)$ with the choice $J(\mathbf{x}) = J_0(\mathbf{x}) \equiv \|\mathbf{x}\|_0$; explicitly,

$$(2) \qquad (P_0): \qquad \min_{\mathbf{x}} \ \|\mathbf{x}\|_0 \ \text{ subject to } \ \mathbf{b} = \mathbf{Ax}.$$

Sparsity optimization (2) looks superficially like the minimum $\ell_2$-norm problem $(P_2)$, but the notational similarity masks some startling differences. The solution to $(P_2)$ is always unique and is readily available through now-standard tools from computational linear algebra. $(P_0)$ has probably been considered to be a possible goal from time to time for many years, but initially it seems to pose many conceptual challenges that have inhibited its widespread study and application. These are rooted in the discrete and discontinuous nature of the $\ell_0$ norm; the standard convex analysis ideas which underpin the analysis of $(P_2)$ do not apply. Many of the most basic questions about $(P_0)$ seem immune to immediate insight:

- Can uniqueness of a solution be claimed? Under what conditions?
- If a candidate solution is available, can we perform a simple test to verify that the solution is actually the global minimizer of $(P_0)$?

Perhaps, in some instances, with very special matrices $\mathbf{A}$ and left-hand sides $\mathbf{b}$, ways to answer such questions are apparent, but for general classes of problem instances $(\mathbf{A}, \mathbf{b})$, such insights initially seem unlikely.

Beyond conceptual issues of uniqueness and verification of solutions, one is easily overwhelmed by the apparent difficulty of solving $(P_0)$. This is a classical problem of combinatorial search; one sweeps exhaustively through all possible sparse subsets, generating corresponding subsystems $\mathbf{b} = \mathbf{A}_S\mathbf{x}_S$, where $\mathbf{A}_S$ denotes the matrix with $|S|$ columns chosen from those columns of $\mathbf{A}$ with indices in $S$, and checking whether $\mathbf{b} = \mathbf{A}_S\mathbf{x}_S$ can be solved. The complexity of exhaustive search is exponential in $m$ and, indeed, it has been proven that $(P_0)$ is, in general, NP-hard [125]. Thus, a mandatory and crucial set of questions arises: Can $(P_0)$ be efficiently solved by some other means? Can approximate solutions be accepted? How accurate can those be? What kind of approximations will work?

In fact, why should anyone believe that any progress of any kind is possible here? Here is a hint. Let $\|\mathbf{x}\|_1$ denote the $\ell_1$ norm $\sum_i |x_i|$, and consider the problem $(P_1)$ obtained by setting $J(\mathbf{x}) = J_1(\mathbf{x}) = \|\mathbf{x}\|_1$. This problem is somehow intermediate between $(P_2)$ and $(P_0)$. It is a convex optimization problem, and among convex problems it is in some sense the one closest to $(P_0)$. We will see below [49, 93, 46] that for matrices $\mathbf{A}$ with incoherent columns, whenever $(P_0)$ has a sufficiently sparse solution, that solution is unique and is equal to the solution of $(P_1)$. Since $(P_1)$ is convex, the solution can thus be obtained by standard optimization tools—in fact, linear programming. Even more surprisingly, for the same class $\mathbf{A}$, some very simple greedy algorithms (GAs) can also find the sparsest solution to $(P_0)$ [156].

Today many pure and applied mathematicians are pursuing results concerning sparse solutions to underdetermined systems of linear equations. The results achieved so far range from identifying conditions under which $(P_0)$ has a unique solution, to conditions under which $(P_0)$ has the same solution as $(P_1)$, to conditions under which the solution can be found by some "pursuit" algorithm. Extensions range even more widely, from less restrictive notions of sparsity to the impact of noise, the behavior of approximate solutions, and the properties of problem instances defined by ensembles of random matrices. We hope to introduce the reader to some of this work below and provide some appropriate pointers to the growing literature.

### 1.2. The Signal Processing Perspective.

We now know that finding sparse solutions to underdetermined linear systems is a better-behaved and much more practically relevant notion than we might have supposed just a few years ago.

In parallel with this development, another insight has been developing in signal and image processing, where it has been found that many media types (still imagery, video, acoustic) can be sparsely represented using transform-domain methods, and in fact many important tasks dealing with such media can be fruitfully viewed as finding sparse solutions to underdetermined systems of linear equations.

Many readers will be familiar with the media encoding standard JPEG and its successor, JPEG-2000 [153]. Both standards are based on the notion of transform encoding. The data vector representing the raw pixel samples are transformed—i.e., represented in a new coordinate system—and the resulting coordinates are then processed to produce the encoded bitstream. JPEG relies on the discrete cosine transform (DCT)—a variant of the Fourier transform—while JPEG-2000 relies on the discrete wavelet transform (DWT) [116]. These transforms can be viewed analytically as rotations of coordinate axes from the standard Euclidean basis to a new basis. Why does it make sense to change coordinates in this way? Sparsity provides the answer.

The DCT of media content often has the property that the *first several* transform coefficients are quite large and later ones are very small. Treating the later coefficients as zeros and approximating the early ones by quantized representations yields an approximate coefficient sequence that can be efficiently stored in a few bits. The approximate coefficient sequence can be inverse transformed to yield an approximate representation of the original media content. The DWT of media content has a slightly different property: there are often *a relatively few large coefficients* (although they are not necessarily the "first" ones). Approximating the DWT by setting to zero the small coefficients and quantizing the large ones yields a sequence to be efficiently stored and later inverse transformed to provide an approximate representation of the original media content. The success of the DWT in image coding is thus directly tied to its ability to sparsify image content. For many types of image content, JPEG-2000 outperforms JPEG: fewer bits are needed for a given accuracy or approximation. One underlying reason is that the DWT of such media is more sparse than the DCT representation.[1]

In short, sparsity of representation is key to widely used techniques of transform-based image compression. Transform sparsity is also a driving factor for other important signal and image processing problems, including image denoising [50, 51, 27, 43, 53, 52, 144, 124, 96] and image deblurring [76, 75, 74, 41]. Repeatedly, it has been shown that a better representation technique—one that leads to more sparsity—can be the basis for a practically better solution to such problems. For example, it has been found that for certain media types (e.g., musical signals with strong harmonic content), sinusoids are best for compression, noise removal, and deblurring, while for other media types (e.g., images with strong edges), wavelets are a better choice than sinusoids.

Realistic media may be a superposition of several such types, conceptually requiring both sinusoids and wavelets. Following this idea leads to the notion of joining together sinusoids and wavelets in a combined representation. Mathematically this now lands us in a situation similar to that described in the previous section. The basis of sinusoids alone makes an $n \times n$ matrix, and the basis of wavelets makes an $n \times n$ matrix; the concatenation makes an $n \times 2n$ matrix. The problem of finding a sparse representation of a signal vector $\mathbf{b}$ using such a system is exactly the same as that of the previous section. We have a system of $n$ equations in $m = 2n$ unknowns, which we know is underdetermined; we look to sparsity as a principle to find the desired solution.

We can make this connection formal as follows. Let $\mathbf{b}$ denote the vector of signal/image values to be represented, and let $\mathbf{A}$ be the matrix whose columns are the elements of the different bases to be used in the representation. The problem $(P_0)$ offers literally the sparsest representation of the signal content.

### 1.3. Measuring Sparsity.

The $\ell_0$ norm, while providing a very simple and easily grasped notion of sparsity, is not the only notion of sparsity available to us, nor is it really the right notion for empirical work. A vector of real data would rarely be representable by a vector of coefficients containing many strict zeros. A weaker notion of sparsity can be built on the notion of approximately representing a vector using a small number of nonzeros; this can be quantified by the weak $\ell_p$ norms, which measure the tradeoff between the number of nonzeros and the $\ell_2$ error of reconstruction. Denoting by $N(\epsilon, \mathbf{x})$ the number of entries in $\mathbf{x}$ exceeding $\epsilon$, these measures of sparsity are defined by

$$\|\mathbf{x}\|_{w\ell_p} = \sup_{\epsilon > 0} \; N(\epsilon, \mathbf{x}) \cdot \epsilon^p.$$

[1] A second important reason for the superiority of JPEG-2000 is its improved bit-plane-based quantization strategy.

Here $0 < p \leq 1$ is the interesting range of $p$, giving a very powerful sparsity constraint. The weak $\ell_p$ norm is a popular measure of sparsity in the mathematical analysis community; models of cartoon images have sparse representations as measured in weak $\ell_p$ [26, 13].

Almost equivalent are the usual $\ell^p$ norms, defined by

$$\|\mathbf{x}\|_p = \left( \sum_i |x_i|^p \right)^{1/p}.$$

These will seem more familiar objects than the weak $\ell_p$ norms, in the range $1 \leq p \leq \infty$; however, for measuring sparsity, $0 < p < 1$ is of most interest.

It would seem very natural, based on our discussion of media sparsity, to attempt to solve a problem of the form

$$(P_p): \qquad \min \|\mathbf{x}\|_p \text{ subject to } \mathbf{Ax} = \mathbf{b}, \tag{3}$$

for example, with $p = 1/2$ or $p = 2/3$. Unfortunately, each choice $0 < p < 1$ leads to a nonconvex optimization problem which is very difficult to solve in general.

At this point, our discussion of the $\ell_0$ norm in section 1.1 can be brought to bear. The $\ell_0$ norm is naturally related to the $\ell_p$ norms with $0 < p < 1$; all are measures of sparsity and, in fact, the $\ell_0$ norm is the limit as $p \to 0$ of the $\ell_p$ norms in the following sense:

$$\|\mathbf{x}\|_0 = \lim_{p \to 0} \|\mathbf{x}\|_p^p = \lim_{p \to 0} \sum_{k=1}^{m} |x_k|^p. \tag{4}$$

Figure 1 presents the behavior of the scalar weight function $|x|^p$—the core of the norm computation—for various values of $p$, showing that as $p$ goes to zero, this measure becomes a count of the nonzeros in $\mathbf{x}$.

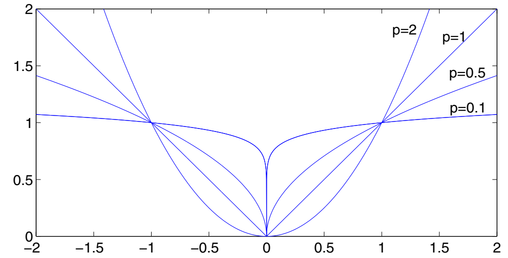


**Fig. 1** *The behavior of $|x|^p$ for various values of $p$. As $p$ tends to zero, $|x|^p$ approaches the indicator function, which is $0$ for $x = 0$ and $1$ elsewhere.*

Note that among the $\ell_p$ norms, the choice $p = 1$ gives a convex functional, while every choice $0 < p < 1$ yields a concave functional. We have already mentioned that solving $(P_0)$ can sometimes be attacked by solving $(P_1)$ instead, or by using an appropriate heuristic GA; the same lesson applies here: although we might want to solve $(P_p)$ we should do better by instead solving $(P_1)$ or by applying an appropriate heuristic GA.

**1.4. This Paper.** The keywords "sparse," "sparsity," "sparse representations," "sparse approximations," and "sparse decompositions" are increasingly popular; the

Institute for Scientific Information in its June 2006 issue of *Essential Science Indicators* formally identified a new research front involving some of the key papers we discuss here. This emerging research front, which for brevity will be referred to hereafter as *Sparse-Land*, can be identified with the topic of sparse modeling, and investigates how to fit models where only a few terms out of many will be used and how to sparsely model important natural data types. In this paper, we hope to give a sampling of some of the work in this new area, spanning the range from theory to applications.

## 2. The Sparsest Solution of $\mathbf{Ax} = \mathbf{b}$.

We return to the basic problem $(P_0)$, which is at the core of our discussion. For the underdetermined linear system of equations $\mathbf{Ax} = \mathbf{b}$ (a full-rank matrix $\mathbf{A} \in \mathbb{R}^{n \times m}$ with $n < m$), we pose the following questions:

Q1: When can uniqueness of the sparsest solution be claimed?
Q2: Can a candidate solution be tested to verify its (global) optimality?
Q3: Can the solution be reliably and efficiently found in practice?
Q4: What performance guarantees can be given for various approximate and practical solvers?

This section addresses all these questions and some of their extensions.

### 2.1. Uniqueness.

#### 2.1.1. Uniqueness via the *Spark*.

A key property that is crucial for the study of uniqueness is the *spark* of the matrix $\mathbf{A}$, a term coined and defined in [46]. We start with the following definition.

DEFINITION 1 (see [46]). *The spark of a given matrix* $\mathbf{A}$ *is the smallest number of columns from* $\mathbf{A}$ *that are linearly dependent.*

Recall that the rank of a matrix is defined as the largest number of columns from $\mathbf{A}$ that are linearly independent. Clearly, the resemblance between these two definitions is noticeable. Nevertheless, the *spark* of a matrix is far more difficult to obtain, compared to the rank, as it calls for a combinatorial search over all possible subsets of columns from $\mathbf{A}$.

The importance of this property of matrices for the study of the uniqueness of sparse solutions was unraveled in [84]. Interestingly, this property previously appeared in the literature of psychometrics (termed *Kruskal rank*), used in the context of studying uniqueness of tensor decomposition [102, 110]. The *spark* is also related to established notions in matroid theory; formally, it is precisely the girth of the linear matroid defined by $\mathbf{A}$, i.e., the length of the shortest cycle in that matroid [162, 6, 61]. Finally, if we consider the same definition where the arithmetic underlying the matrix product is performed *not* over the fields of real or complex numbers but instead over the ring of integers mod $q$, the same quantity arises in coding theory, where it allows the computation of the minimum distance of a code [131]. The resemblance between all these concepts is striking and instructive.

The *spark* gives a simple criterion for uniqueness of sparse solutions. By definition, the vectors in the null-space of the matrix $\mathbf{Ax} = 0$ must satisfy $\|\mathbf{x}\|_0 \geq spark(\mathbf{A})$, since these vectors linearly combine columns from $\mathbf{A}$ to give the zero vector, and at least *spark* such columns are necessary by definition. Using the *spark* we obtain the following result.

THEOREM 2 (uniqueness: *spark* [84, 46]). *If a system of linear equations* $\mathbf{Ax} = \mathbf{b}$ *has a solution* $\mathbf{x}$ *obeying* $\|\mathbf{x}\|_0 < spark(\mathbf{A})/2$, *this solution is necessarily the sparsest possible.*

*Proof.* Consider an alternative solution $\mathbf{y}$ that satisfies the same linear system $\mathbf{Ay} = \mathbf{b}$. This implies that $\mathbf{x} - \mathbf{y}$ must be in the null-space of $\mathbf{A}$, i.e., $\mathbf{A}(\mathbf{x} - \mathbf{y}) = 0$.

By the definition of *spark*,

$$\|\mathbf{x}\|_0 + \|\mathbf{y}\|_0 \geq \|\mathbf{x} - \mathbf{y}\|_0 \geq spark(\mathbf{A}). \tag{5}$$

The leftmost term in the above inequality simply states that the number of nonzeros in the difference vector $\mathbf{x} - \mathbf{y}$ cannot exceed the sum of the number of nonzeros within each of the vectors $\mathbf{x}$ and $\mathbf{y}$ separately. Since we have a solution satisfying $\|\mathbf{x}\|_0 < spark(\mathbf{A})/2$, we conclude that any alternative solution $\mathbf{y}$ necessarily has more than $spark(\mathbf{A})/2$ nonzeros, as claimed. □

This result is very elementary and yet quite surprising, bearing in mind that $(P_0)$ is a highly complicated optimization task of combinatorial flavor. In general combinatorial optimization problems, when considering a proposed solution, one hopes only to check local optimality—i.e., that no simple modification gives a better result. Here, we find that simply checking the solution sparsity, and comparing that with the *spark*, lets us check global optimality.

Clearly, the value of *spark* can be very informative, and large values of *spark* are evidently very useful. How large can *spark* be? By definition, *spark* must be in the range $1 \leq spark(\mathbf{A}) \leq n + 1$. For example, if $\mathbf{A}$ comprises random independent and identically distributed (i.i.d.) entries (say, Gaussian), then with probability 1 we have $spark(\mathbf{A}) = n + 1$, implying that no $n$ columns are linearly dependent. In this case, uniqueness is ensured for every solution with $n/2$ or fewer nonzero entries.

### 2.1.2. Uniqueness via the Mutual Coherence.

The *spark* is at least as difficult to evaluate as solving $(P_0)$. Thus, simpler ways to guarantee uniqueness are of interest. A very simple way exploits the *mutual coherence* of the matrix $\mathbf{A}$, defined as follows.

DEFINITION 3 (see [118, 49, 46]). *The* mutual coherence *of a given matrix* $\mathbf{A}$ *is the largest absolute normalized inner product between different columns from* $\mathbf{A}$*. Denoting the kth column in* $\mathbf{A}$ *by* $\mathbf{a}_k$*, the* mutual coherence *is given by*

$$\mu(\mathbf{A}) = \max_{1 \leq k,j \leq m,\ k \neq j} \frac{\left|\mathbf{a}_k^T \mathbf{a}_j\right|}{\left\|\mathbf{a}_k\right\|_2 \cdot \left\|\mathbf{a}_j\right\|_2}. \tag{6}$$

The mutual coherence is a way to characterize the dependence between columns of the matrix $\mathbf{A}$. For a unitary matrix, columns are pairwise orthogonal, and so the mutual coherence is zero. For general matrices with more columns than rows, $m > n$, $\mu$ is necessarily strictly positive, and we desire the smallest possible value so as to get as close as possible to the behavior exhibited by unitary matrices.

The work reported in [49, 93] considered structured matrices $\mathbf{A} \in \mathbb{R}^{n \times 2n} = [\mathbf{\Phi}\ \mathbf{\Psi}]$, where $\mathbf{\Phi}$ and $\mathbf{\Psi}$ are unitary matrices. The mutual coherence of such dictionaries satisfies $1/\sqrt{n} \leq \mu(\mathbf{A}) \leq 1$, where the lower bound is achievable for certain pairs of orthogonal bases, such as the identity and the Fourier, the identity and the Hadamard, and so on. When considering random orthogonal matrices of size $n \times m$, the work in [49, 93] has shown that they tend to be incoherent, implying that $\mu(\mathbf{A}_{n,m})$ is typically proportional to $\sqrt{\log(nm)/n}$ for $n \to \infty$. In [150] it has been shown that for full-rank matrices of size $n \times m$ the mutual coherence is bounded from below by

$$\mu \geq \sqrt{\frac{m - n}{n(m - 1)}}\ ,$$

equality being obtained for a family of matrices named *Grassmanian frames*. Indeed, this family of matrices has $spark(\mathbf{A}) = n + 1$, the highest value possible. We also mention work in quantum information theory, constructing error-correcting codes using a collection of orthogonal bases with minimal coherence, obtaining similar bounds on the mutual coherence for amalgams of orthogonal bases [11].

Mutual coherence, relatively easy to compute, allows us to lower bound the *spark*, which is often hard to compute.

LEMMA 4 (see [46]). *For any matrix* $\mathbf{A} \in \mathbb{R}^{n\times m}$*, the following relationship holds:*

$$spark(\mathbf{A}) \geq 1 + \frac{1}{\mu(\mathbf{A})}. \tag{7}$$

*Proof.* First, modify the matrix $\mathbf{A}$ by normalizing its columns to unit $\ell_2$ norm, obtaining $\tilde{\mathbf{A}}$. This operation preserves both the *spark* and the mutual coherence. The entries of the resulting Gram matrix $\mathbf{G} = \tilde{\mathbf{A}}^T\tilde{\mathbf{A}}$ satisfy the following properties:

$$\{G_{k,k} = 1 \ : \ 1 \leq k \leq m\} \quad \text{and} \quad \{|G_{k,j}| \leq \mu \ : \ 1 \leq k, j \leq m, \ k \neq j\}.$$

Consider an arbitrary minor from $\mathbf{G}$ of size $p \times p$, built by choosing a subgroup of $p$ columns from $\tilde{\mathbf{A}}$ and computing their sub-Gram matrix. From the Gershgorin disk theorem [91], if this minor is diagonally dominant—i.e., if $\sum_{j\neq i} |G_{i,j}| < |G_{i,i}|$ for every $i$—then this submatrix of $\mathbf{G}$ is positive definite, and so those $p$ columns from $\tilde{\mathbf{A}}$ are linearly independent. The condition $p < 1 + 1/\mu$ implies positive definiteness of every $p \times p$ minor, and so $spark(\mathbf{A}) \geq p + 1 \geq 1 + 1/\mu$. □

We have the following analogue of Theorem 2.

THEOREM 5 (uniqueness: mutual coherence [46]). *If a system of linear equations* $\mathbf{Ax} = \mathbf{b}$ *has a solution* $\mathbf{x}$ *obeying* $\|\mathbf{x}\|_0 < \frac{1}{2}(1 + 1/\mu(\mathbf{A}))$*, this solution is necessarily the sparsest possible.*

Compare Theorems 2 and 5. They are parallel in form, but with different assumptions. In general, Theorem 2, which uses *spark*, is sharp and far more powerful than Theorem 5, which uses the coherence and so only a lower bound on *spark*. The coherence can never be smaller than $1/\sqrt{n}$, and, therefore, the cardinality bound of Theorem 5 is never larger than $\sqrt{n}/2$. However, the *spark* can easily be as large as $n$, and Theorem 2 then gives a bound as large as $n/2$.

We have now given partial answers to the questions Q1 and Q2 posed at the start of this section. We have seen that any sufficiently sparse solution is guaranteed to be unique among sufficiently sparse solutions. Consequently, any sufficiently sparse solution is necessarily the global optimizer of $(P_0)$. These results show that searching for a sparse solution can lead to a well-posed question with interesting properties. We now turn to discuss Q3—practical methods for obtaining solutions.

### 2.2. Pursuit Algorithms: Practice.

A straightforward approach to solving $(P_0)$ seems hopeless; we now discuss methods which, it seems, have no hope of working—but which, under specific conditions, will work.

#### 2.2.1. Greedy Algorithms.

Suppose that the matrix $\mathbf{A}$ has $spark(\mathbf{A}) > 2$ and the optimization problem $(P_0)$ has value $val(P_0) = 1$, so $\mathbf{b}$ is a scalar multiple of some column of the matrix $\mathbf{A}$. We can identify this column by applying $m$ tests—one per column of $\mathbf{A}$. This procedure requires order $O(mn)$ flops, which may be considered reasonable. Now suppose that $\mathbf{A}$ has $spark(\mathbf{A}) > 2k_0$, and the optimization problem is known to have value $val(P_0) = k_0$. Then $\mathbf{b}$ is a linear combination of at most $k_0$ columns of $\mathbf{A}$. Generalizing the previous solution, one might try to enumerate all $\binom{m}{k_0} = O(m^{k_0})$ subsets of $k_0$ columns from $\mathbf{A}$ and then to test each one. Enumeration takes $O(m^{k_0} n {k_0}^2)$ flops, which seems prohibitively slow in many settings.

A greedy strategy abandons exhaustive search in favor of a series of locally optimal single-term updates. Starting from $\mathbf{x}^0 = 0$ it iteratively constructs a $k$-term approximant $\mathbf{x}^k$ by maintaining a set of active columns—initially empty—and, at each stage, expanding that set by one additional column. The column chosen at each stage maximally reduces the residual $\ell_2$ error in approximating $\mathbf{b}$ from the currently active columns. After constructing an approximant including the new column, the residual $\ell_2$ error is evaluated; if it now falls below a specified threshold, the algorithm terminates.

Exhibit 1 presents a formal description of the strategy and its associated notation. This procedure is known in the literature of signal processing by the name *orthogonal matching pursuit* (OMP), but is very well known (and was used much earlier) by other names in other fields—see below.

**Task:** Approximate the solution of $(P_0)$: $\min_{\mathbf{x}} \ \|\mathbf{x}\|_0$ subject to $\mathbf{A}\mathbf{x} = \mathbf{b}$.

**Parameters:** We are given the matrix $\mathbf{A}$, the vector $\mathbf{b}$, and the error threshold $\epsilon_0$.

**Initialization:** Initialize $k = 0$, and set

- The initial solution $\mathbf{x}^0 = 0$.
- The initial residual $\mathbf{r}^0 = \mathbf{b} - \mathbf{A}\mathbf{x}^0 = \mathbf{b}$.
- The initial solution support $\mathcal{S}^0 = Support\{\mathbf{x}^0\} = \emptyset$.

**Main Iteration:** Increment $k$ by 1 and perform the following steps:

- **Sweep:** Compute the errors $\epsilon(j) = \min_{z_j} \|\mathbf{a}_j z_j - \mathbf{r}^{k-1}\|_2^2$ for all $j$ using the optimal choice $z_j^* = \mathbf{a}_j^T \mathbf{r}^{k-1} / \|\mathbf{a}_j\|_2^2$.
- **Update Support:** Find a minimizer $j_0$ of $\epsilon(j)$: $\forall \ j \notin \mathcal{S}^{k-1}, \ \epsilon(j_0) \le \epsilon(j)$, and update $\mathcal{S}^k = \mathcal{S}^{k-1} \cup \{j_0\}$.
- **Update Provisional Solution:** Compute $\mathbf{x}^k$, the minimizer of $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$ subject to $Support\{\mathbf{x}\} = \mathcal{S}^k$.
- **Update Residual:** Compute $\mathbf{r}^k = \mathbf{b} - \mathbf{A}\mathbf{x}^k$.
- **Stopping Rule:** If $\|\mathbf{r}^k\|_2 < \epsilon_0$, stop. Otherwise, apply another iteration.

**Output:** The proposed solution is $\mathbf{x}^k$ obtained after $k$ iterations.

**Exhibit 1.** *OMP—a GA for approximating the solution of* $(P_0)$*.*

If the approximation delivered has $k_0$ nonzeros, the method requires $\mathcal{O}(k_0 mn)$ flops in general; this can be dramatically better than the exhaustive search, which requires $\mathcal{O}(nm^{k_0} {k_0}^2)$ flops.

Thus, this single-term-at-a-time strategy can be much more efficient than exhaustive search—if it works! The strategy can fail badly, i.e., there are explicit examples (see [154, 155, 36]) where a simple $k$-term representation is possible, but this approach yields an $n$-term (i.e., dense) representation. In general, all that can be said is that among single-term-at-a-time strategies, the approximation error is always reduced by as much as possible, given the starting approximation and the single-term-at-a-time constraint. This explains why this type of algorithm has earned the name "greedy algorithm" in approximation theory.

Many variants on this algorithm are available, offering improvements in accuracy or in complexity or both [118, 34, 33, 23, 130, 30, 159, 82]. This family of GAs is well known and extensively used, and, in fact, these algorithms have been reinvented in various fields. In the setting of statistical modeling, greedy stepwise least squares is called *forward stepwise regression* and has been widely used since at least the 1960s [31, 90]. When used in the signal processing setting this goes by the name of *matching pursuit* (MP) [118, 34, 33] or OMP [23, 130]. Approximation theorists refer to these algorithms as GAs and consider several variants of them—the pure (PGA), the orthogonal (OGA), the relaxed (RGA), and the weak GA (WGA) [154, 155, 36, 4, 87].

**2.2.2. Convex Relaxation Techniques.** A second way to render $(P_0)$ more tractable is to regularize the (highly discontinuous) $\ell_0$ norm, replacing it by a continuous or even smooth approximation. Examples of such regularizations include replacing it with $\ell_p$ norms for some $p \in (0,1]$ or even by smooth functions such as $\sum_j \log(1+\alpha x_j^2)$ or $\sum_j x_j^2/(\alpha+x_j^2)$. As an example, the FOCUSS method [84, 139, 138] uses $\ell_p$ for some fixed $p \in (0,1]$ and seeks a local minimum of the $\ell_p$ norm by iteratively reweighted least squares [97]. This is a practical strategy, but little is known about circumstances where it will be successful, i.e., when a numerical local minimum will actually be a good approximation to a global minimum of $(P_0)$. Another strategy is to replace the $\ell_0$ norm by the $\ell_1$ norm, which is, in a natural sense, its best convex approximant [24, 25, 142]; many optimization tools are available "off the shelf" for solving $(P_1)$.

Turning from $(P_0)$ to its regularizations $(P_p)$ with $0 < p \leq 1$, care must be taken with respect to normalization of the columns in $\mathbf{A}$. While the $\ell_0$ norm is indifferent to the magnitude of the nonzero entries in $\mathbf{x}$, the $\ell_p$ norms tend to penalize higher magnitudes and thus bias the solution toward choosing to put nonzero entries in $\mathbf{x}$ in locations that multiply large norm columns in $\mathbf{A}$. In order to avoid this bias, the columns should be scaled appropriately.

Convexifying with the $\ell_1$ norm, the new objective becomes

$$(P_1): \quad \min_{\mathbf{x}} \quad \|\mathbf{W}\mathbf{x}\|_1 \text{ subject to } \mathbf{b} = \mathbf{A}\mathbf{x}. \tag{8}$$

The matrix $\mathbf{W}$ is a diagonal positive-definite matrix that introduces the above-described precompensating weights. A natural choice for the $(i,i)$ entry in this matrix for this case is $w(i) = \|\mathbf{a}_i\|_2$. Assuming that $\mathbf{A}$ has no zero columns, all these norms are strictly positive and the problem $(P_1)$ is well defined. The case where all the columns of $\mathbf{A}$ are normalized (and thus $\mathbf{W} = \mathbf{I}$) was named *basis pursuit* (BP) in [24]. We will use this name hereafter for the more general setup in (8).

The problem $(P_1)$ can be cast as a linear programming (LP) problem and solved using modern interior-point methods, simplex methods, or other techniques, such as homotopy methods [24]. Such algorithms are far more sophisticated than the GAs mentioned earlier, as they obtain the global solution of a well-defined optimization problem. This also makes it possible to understand their working in sometimes great detail—something which is apparently not the case for GAs.

While there are several readily available and carefully programmed solvers for accurate solution of $(P_1)$, approximating the solution of $(P_0)$ by GAs is still considered to be more common, perhaps because of the perception that high-accuracy solution of $(P_1)$ is a computationally heavy task. An emerging alternative to approximately solving $(P_1)$ has been recently introduced in several independent papers, leading to similar algorithms that may be called *iterated shrinkage* methods [32, 76, 75, 5, 62, 67]. These iteratively use multiplication by $\mathbf{A}$ and its adjoint, and a simple 1D operation that sets to zero small entries—a *shrinkage* operation. These methods can compete with the greedy methods in simplicity and efficiency. However, this line of work is still in its infancy, more work being required to establish the effectiveness of such algorithms compared to the greedy ones. More on this family of techniques is presented in section 3.2.3.

### 2.3. Pursuit Algorithms: Performance.

So far we have provided answers to Q1–Q3. We now discuss Q4—performance guarantees of the above-described pursuit algorithms.

Assume that the linear system $\mathbf{Ax} = \mathbf{b}$ has a sparse solution with $k_0$ nonzeros, i.e., $\|\mathbf{x}\|_0 = k_0$. Furthermore, assume that $k_0 < spark(\mathbf{A})/2$. Will MP or BP succeed in recovering the sparsest solution? Clearly, such success cannot be expected for all $k_0$ and for all matrices $\mathbf{A}$, since this would conflict with the known NP-hardness of the problem in the general case. However, if the equation actually has a "sufficiently sparse" solution, the success of these algorithms in addressing the original objective $(P_0)$ can be guaranteed [46, 86, 79, 156, 133, 134, 135]. We present here two such results, one that corresponds to the OMP algorithm as described in Exhibit 1, and the other for BP (i.e., solving $(P_1)$ in place of $(P_0)$).

#### 2.3.1. The GA Solves $(P_0)$ in Sufficiently Sparse Cases.

Theorem 6 (equivalence: OGA [156, 48]). *For a system of linear equations* $\mathbf{Ax} = \mathbf{b}$ *(*$\mathbf{A} \in \mathbb{R}^{n\times m}$ *full-rank with* $n < m$*), if a solution* $\mathbf{x}$ *exists obeying*

$$\|\mathbf{x}\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{A})}\right), \tag{9}$$

*an OGA run with threshold parameter* $\epsilon_0 = 0$ *is guaranteed to find it exactly.*

*Proof.* Suppose, without loss of generality, that the sparsest solution of the linear system is such that all its $k_0$ nonzero entries are at the beginning of the vector, in decreasing order of the values $|x_j| \cdot \|\mathbf{a}_j\|_2$. Thus,

$$\mathbf{b} = \mathbf{Ax} = \sum_{t=1}^{k_0} x_t \mathbf{a}_t. \tag{10}$$

At the first step ($k = 0$) of the algorithm, $\mathbf{r}^k = \mathbf{r}^0 = \mathbf{b}$, and the set of computed errors from the **Sweep** step are given by

$$\epsilon(j) = \min_{z_j} \|\mathbf{a}_j z_j - \mathbf{b}\|_2^2 = \left\|\mathbf{a}_j \frac{\mathbf{a}_j^T\mathbf{b}}{\|\mathbf{a}_j\|_2^2} - \mathbf{b}\right\|_2^2 = \|\mathbf{b}\|_2^2 - \frac{(\mathbf{a}_j^T\mathbf{b})^2}{\|\mathbf{a}_j\|_2^2} \geq 0.$$

Thus, for the first step to choose one of the first $k_0$ entries in the vector (and thus do well), we must require that for all $i > k_0$ (columns outside the true support),

$$\left|\frac{\mathbf{a}_1^T\mathbf{b}}{\|\mathbf{a}_1\|_2}\right| > \left|\frac{\mathbf{a}_i^T\mathbf{b}}{\|\mathbf{a}_i\|_2}\right|. \tag{11}$$

Substituting this in (10), this requirement translates into

$$\left|\sum_{t=1}^{k_0} x_t \frac{\mathbf{a}_1^T\mathbf{a}_t}{\|\mathbf{a}_1\|_2}\right| > \left|\sum_{t=1}^{k_0} x_t \frac{\mathbf{a}_i^T\mathbf{a}_t}{\|\mathbf{a}_i\|_2}\right|. \tag{12}$$

In order to consider the worst-case scenario, we should construct a lower bound for the left-hand side, an upper bound for the right-hand side, and then pose the above requirement again. For the left-hand side we have

$$\begin{aligned}\left|\sum_{t=1}^{k_0} x_t \frac{\mathbf{a}_1^T \mathbf{a}_t}{\|\mathbf{a}_1\|_2}\right| &\geq |x_1| \cdot \|\mathbf{a}_1\|_2 - \sum_{t=2}^{k_0} |x_t| \cdot \left|\frac{\mathbf{a}_1^T \mathbf{a}_t}{\|\mathbf{a}_1\|_2}\right| \\ &\geq |x_1| \cdot \|\mathbf{a}_1\|_2 - \sum_{t=2}^{k_0} |x_t| \cdot \|\mathbf{a}_t\|_2 \cdot \mu(\mathbf{A}) \qquad (13) \\ &\geq |x_1| \cdot \|\mathbf{a}_1\|_2 \left(1 - \mu(\mathbf{A})(k_0 - 1)\right).\end{aligned}$$

Here we have exploited the definition of the mutual coherence $\mu(\mathbf{A})$ in (6) and the descending ordering of the values $|x_j| \cdot \|\mathbf{a}_j\|_2$. Similarly, the right-hand-side term is bounded by

$$\begin{aligned}\left|\sum_{t=1}^{k_0} x_t \frac{\mathbf{a}_i^T \mathbf{a}_t}{\|\mathbf{a}_i\|_2}\right| &\leq \sum_{t=1}^{k_0} |x_t| \cdot \left|\frac{\mathbf{a}_i^T \mathbf{a}_t}{\|\mathbf{a}_i\|_2}\right| \\ &\leq \sum_{t=1}^{k_0} |x_t| \cdot \|\mathbf{a}_t\|_2 \cdot \mu(\mathbf{A}) \qquad (14) \\ &\leq |x_1| \cdot \|\mathbf{a}_1\|_2 \cdot \mu(\mathbf{A}) k_0.\end{aligned}$$

Using these two bounds plugged into the inequality in (12), we obtain

$$\begin{aligned}\left|\sum_{t=1}^{k_0} x_t \frac{\mathbf{a}_1^T \mathbf{a}_t}{\|\mathbf{a}_1\|_2}\right| &\geq |x_1| \cdot \|\mathbf{a}_1\|_2 \left(1 - \mu(\mathbf{A})(k_0 - 1)\right) \qquad (15) \\ &> |x_1| \cdot \|\mathbf{a}_1\|_2 \mu(\mathbf{A}) k_0 \geq \left|\sum_{t=1}^{k_0} x_t \frac{\mathbf{a}_i^T \mathbf{a}_t}{\|\mathbf{a}_i\|_2}\right|,\end{aligned}$$

which leads to

$$1 + \mu(\mathbf{A}) > 2\mu(\mathbf{A}) k_0 \quad \Rightarrow \quad k_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{A})}\right), \qquad (16)$$

which is exactly the condition of sparsity required in Theorem 6. This condition guarantees the success of the first stage of the algorithm, implying that the chosen element must be in the correct support of the sparsest decomposition. Once done, the next step is an update of the solution and the residual $\mathbf{r}^1$. This residual can be written as

$$\mathbf{r}^1 = \mathbf{b} - \mathbf{a}_k z_k^* = \sum_{t=1}^{k_0} \tilde{x}_t \mathbf{a}_t, \qquad (17)$$

where $1 \leq k \leq k_0$, and the value of $z_k^*$ is such that $\mathbf{r}^1$ is orthogonal to $\mathbf{a}_k$ due to the least-squares computation. Repeating the above process, we can assume, without loss of generality, that the entries of $\tilde{\mathbf{x}}$ have been rearranged in decreasing order of the values $|\tilde{x}_j| \cdot \|\mathbf{a}_j\|_2$ by a simple permutation of the columns in $\mathbf{A}$. Using the same set of steps we obtain that condition (16) guarantees that the algorithm again finds an index from the true support of the solution. Indeed, due to the orthogonality $\mathbf{a}_k^T \mathbf{r}^k = 0$ we necessarily find that $\epsilon(k)$ is the highest among the computed errors, and as such will not be chosen again.

Repeating this reasoning, the same holds true for $k_0$ such iterations—hence the algorithm always selects values from the correct set of indices, and always an index that has not been chosen yet. After $k_0$ such iterations, the residual becomes zero and the algorithm stops, ensuring the success of the overall algorithm in recovering the correct solution $\mathbf{x}$ as the theorem claims. □

**2.3.2. Basis Pursuit Solves ($P_0$) in Sufficiently Sparse Cases.** We now turn to consider BP, i.e., the replacement of $(P_0)$ by $(P_1)$ as an optimization problem.

THEOREM 7 (equivalence: BP [46, 86]). *For the system of linear equations* $\mathbf{Ax} = \mathbf{b}$ *(*$\mathbf{A} \in \mathbb{R}^{n\times m}$ *full-rank with* $n < m$*), if a solution* $\mathbf{x}$ *exists obeying*

$$\|\mathbf{x}\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{A})}\right), \tag{18}$$

*that solution is both the unique solution of* $(P_1)$ *and the unique solution of* $(P_0)$*.*

Note that the assumptions of the theorem concerning BP are the same as those of Theorem 6 concerning MP. This does not mean that the two algorithms are always expected to perform similarly! Empirical evidence will be presented in section 3.3.1 showing that these two methods often behave differently.

*Proof.* Define the following set of alternative solutions:

$$\mathcal{C} = \{\mathbf{y} \mid \mathbf{y} \neq \mathbf{x}, \quad \|\mathbf{Wy}\|_1 \leq \|\mathbf{Wx}\|_1, \quad \|\mathbf{y}\|_0 > \|\mathbf{x}\|_0, \quad \text{and} \quad \mathbf{A}(\mathbf{y}-\mathbf{x}) = 0\}. \tag{19}$$

This set contains all the possible solutions that are different from $\mathbf{x}$, have larger support, satisfy the linear system of equations $\mathbf{Ay} = \mathbf{b}$, and are at least as good from the weighted $\ell_1$ perspective. This set being nonempty implies that there is an alternative solution that the BP will find, rather than the desired $\mathbf{x}$.

In view of Theorem 5 and the fact that $\|\mathbf{x}\|_0 < (1 + 1/\mu(\mathbf{A}))/2$, $\mathbf{x}$ is necessarily the unique sparsest possible solution, and hence alternative solutions ($\mathbf{y} \neq \mathbf{x}$) are necessarily "denser." Thus, this requirement can be omitted from the definition of $\mathcal{C}$. Defining $\mathbf{e} = \mathbf{y} - \mathbf{x}$, we can rewrite $\mathcal{C}$ as a shifted version around $\mathbf{x}$,

$$\mathcal{C}_s = \{\mathbf{e} \mid \mathbf{e} \neq 0, \quad \|\mathbf{W}(\mathbf{e}+\mathbf{x})\|_1 - \|\mathbf{Wx}\|_1 \leq 0, \quad \text{and} \quad \mathbf{Ae} = 0\}. \tag{20}$$

The strategy of the proof we are about to present is to enlarge this set and show that even this enlarged set is empty. This will prove that BP indeed succeeds in recovering $\mathbf{x}$.

We start with the requirement that $\|\mathbf{W}(\mathbf{e}+\mathbf{x})\|_1 - \|\mathbf{Wx}\|_1 \leq 0$. Assuming, without loss of generality, that by a simple column permutation of $\mathbf{A}$, the $k_0$ nonzeros in $\mathbf{x}$ are located at the beginning of the vector, this requirement can be written as

$$\|\mathbf{W}(\mathbf{e}+\mathbf{x})\|_1 - \|\mathbf{Wx}\|_1 = \sum_{j=1}^{k_0} w(j)\cdot(|e_j + x_j| - |x_j|) + \sum_{j>k_0} w(j)\cdot|e_j| \leq 0. \tag{21}$$

Using the inequality $|a+b| - |b| \geq -|a|$ and the fact that $w(i) > 0$, we can relax the above condition and demand instead that

$$\begin{aligned} -\sum_{j=1}^{k_0} w(j)\cdot|e_j| + \sum_{j>k_0} w(j)\cdot|e_j| &\leq \sum_{j=1}^{k_0} w(j)\cdot(|e_j + x_j| - |x_j|) \\ &\quad + \sum_{j>k_0} w(j)\cdot|e_j| \leq 0. \end{aligned} \tag{22}$$

This inequality can be written more compactly by adding and subtracting the term $\sum_{j=1}^{k_0} w(j) \cdot |e_j|$ and denoting it as $\mathbf{1}_{k_0}^T \cdot |\mathbf{We}|$ to indicate that it is a sum of the first $k_0$ entries of the vector $|\mathbf{We}|$. This leads to

$$\|\mathbf{We}\|_1 - 2\mathbf{1}_{k_0}^T \cdot |\mathbf{We}| \leq 0. \tag{23}$$

Thus, substituting into the definition of $\mathcal{C}_s$ we get

$$\mathcal{C}_s \subseteq \left\{\mathbf{e} \;\middle|\; \mathbf{e} \neq 0, \quad \|\mathbf{We}\|_1 - 2\mathbf{1}_{k_0}^T \cdot |\mathbf{We}| \leq 0, \quad \text{and} \;\; \mathbf{Ae} = 0\right\} = \mathcal{C}_s^1. \tag{24}$$

We now turn to handling the requirement $\mathbf{Ae} = 0$, replacing it with a relaxed requirement that expands the set $\mathcal{C}_s^1$ further. First, a multiplication by $\mathbf{A}^T$ yields the condition $\mathbf{A}^T\mathbf{Ae} = 0$, which does not yet change the set $\mathcal{C}_s^1$. This equation can be rewritten as

$$\mathbf{W}^{-1}\mathbf{A}^T\mathbf{AW}^{-1}\mathbf{We} = 0. \tag{25}$$

The left multiplication by the inverse of $\mathbf{W}$ leaves the condition the same. The inner multiplication by $\mathbf{W}$ and its inverse cancel out. The term $\mathbf{W}^{-1}\mathbf{A}^T\mathbf{AW}^{-1}$ is desirable, since every entry in this matrix is the normalized inner product used for the definition of the mutual coherence $\mu(\mathbf{A})$. Also, the main diagonal of this matrix contains ones. Thus, (25) can be rewritten by adding and removing $\mathbf{We}$, as follows:

$$-\mathbf{We} = (\mathbf{W}^{-1}\mathbf{A}^T\mathbf{AW}^{-1} - \mathbf{I})\mathbf{We}. \tag{26}$$

Taking an entrywise absolute value on both sides we relax the requirement on $\mathbf{e}$ and obtain

$$\begin{aligned} |\mathbf{We}| = |(\mathbf{W}^{-1}\mathbf{A}^T\mathbf{AW}^{-1} - \mathbf{I})\mathbf{We}| &\leq |\mathbf{W}^{-1}\mathbf{A}^T\mathbf{AW}^{-1} - \mathbf{I}| \cdot |\mathbf{We}| \\ &\leq \mu(\mathbf{A})(\mathbf{1} - \mathbf{I}) \cdot |\mathbf{We}|. \end{aligned} \tag{27}$$

The term $\mathbf{1}$ stands for a rank-1 matrix filled with ones. In the last step above we used the definition of the mutual coherence and the fact that it bounds from above all normalized inner products of the columns of $\mathbf{A}$. Returning to the set $\mathcal{C}_s^1$, we can write

$$\mathcal{C}_s^1 \subseteq \left\{\mathbf{e} \;\middle|\; \mathbf{e} \neq 0,\; \|\mathbf{We}\|_1 - 2\mathbf{1}_{k_0}^T \cdot |\mathbf{We}| \leq 0, \;\text{and}\;\; |\mathbf{We}| \leq \frac{\mu(\mathbf{A})}{1+\mu(\mathbf{A})}\mathbf{1} \cdot |\mathbf{We}|\right\} = \mathcal{C}_s^2. \tag{28}$$

Defining $\mathbf{f} = \mathbf{We}$, (28) can be written differently as

$$\mathcal{C}_f = \left\{\mathbf{f} \;\middle|\; \mathbf{f} \neq 0,\; \|\mathbf{f}\|_1 - 2\mathbf{1}_{k_0}^T \cdot |\mathbf{f}| \leq 0, \;\text{and}\;\; |\mathbf{f}| \leq \frac{\mu(\mathbf{A})}{1+\mu(\mathbf{A})}\mathbf{1} \cdot |\mathbf{f}|\right\}. \tag{29}$$

The obtained set $\mathcal{C}_f$ is unbounded since if $\mathbf{f} \in \mathcal{C}_f$, then $\alpha\mathbf{f} \in \mathcal{C}_f$ for all $\alpha \neq 0$. Thus, in order to study its behavior, we can restrict our quest for normalized vectors, $\|\mathbf{f}\|_1 = 1$. This new set, denoted as $\mathcal{C}_r$, becomes

$$\mathcal{C}_r = \left\{\mathbf{f} \;\middle|\; \|\mathbf{f}\|_1 = 1,\; 1 - 2\mathbf{1}_{k_0}^T \cdot |\mathbf{f}| \leq 0, \;\text{and}\;\; |\mathbf{f}| \leq \frac{\mu(\mathbf{A})}{1+\mu(\mathbf{A})}\mathbf{1}\right\}. \tag{30}$$

In the last condition we have used the relation $\mathbf{1}|\mathbf{f}| = \mathbf{1} \cdot \mathbf{1}^T|\mathbf{f}|$ and the fact that $\mathbf{1}^T|\mathbf{f}| = \|\mathbf{f}\|_1 = 1$.

In order for the vector $\mathbf{f}$ to satisfy the requirement $1 - 2\mathbf{1}_{k_0}^T \cdot |\mathbf{f}| \leq 0$, one needs to concentrate its energy in its first $k_0$ entries. However, the requirements $\|\mathbf{f}\|_1 = 1$ and $|f_j| \leq \mu(\mathbf{A})/(1+\mu(\mathbf{A}))$ restrict these $k_0$ entries to be exactly $|f_j| = \mu(\mathbf{A})/(1+\mu(\mathbf{A}))$, because these are the maximal allowed values. Thus, returning to the first condition we get the requirement

$$1 - 2\mathbf{1}_{k_0}^T \cdot |\mathbf{f}| = 1 - 2k_0 \frac{\mu(\mathbf{A})}{1+\mu(\mathbf{A})} \leq 0. \tag{31}$$

This means that if $k_0$ is less than $(1 + 1/\mu(\mathbf{A}))/2$, the set will be necessarily empty, hence implying that BP leads to the desired solution as the theorem claims. □

The above proof amounts to showing that if there are two solutions to an incoherent underdetermined system, one of them being sparse, moving along the line segment between the two solutions causes an increase in the $\ell_1$ norm as we move away from the sparse solution.

Historically, Theorem 7 was found before the OMP result in Theorem 6—it provided a kind of existence proof that something interesting could be said about solving underdetermined systems of equations under some conditions. The fact that the same form of assumptions leads to the same form of results for both algorithms is tantalizing: Is there some deeper meaning? Keep reading!

## 3. Variations on ($P_0$).

So far, we have focused narrowly on a viewpoint which efficiently showed that there was something interesting to be said about finding sparse solutions to underdetermined systems. We now broaden our viewpoint, expanding the connections to a broad and rapidly growing literature. This leads us to some interesting variations on the $(P_0)$-based results discussed so far.

### 3.1. Uncertainty Principles and Sparsity.

The phenomena exposed so far were first noticed in the following concrete setting: the case where $\mathbf{A}$ is the concatenation of two orthogonal matrices, namely, the identity matrix and the Fourier matrix, $\mathbf{A} = [\mathbf{I}\ \mathbf{F}]$. In that setting, the fact that the system $\mathbf{b} = \mathbf{Ax}$ is underdetermined means, concretely, that there are many ways of representing a given signal $\mathbf{b}$ as a superposition of spikes (i.e., columns from the identity matrix) and sinusoids (i.e., columns from the Fourier matrix). A sparse solution of such a system is a representation of that signal as a superposition of a few sinusoids and a few spikes. The uniqueness of such a sparse solution, and the ability of $\ell_1$ minimization to find it, seemed surprising when first noticed.

The first proof, while sharing some key ideas with the proofs given here, was interpreted at the time as a kind of uncertainty principle. As the reader no doubt knows, the classical uncertainty principle says that a signal cannot be tightly concentrated both in time and in frequency, and it places a lower bound on the product of the spread in time and the spread in frequency. The uniqueness of sparse representation for such time-frequency systems $\mathbf{A} = [\mathbf{I}\ \mathbf{F}]$ can be interpreted as saying that a signal cannot be sparsely represented both in time and in frequency. This viewpoint is helpful for understanding some of the preceding abstract discussion, so we briefly develop it here.

#### 3.1.1. Uncertainty Principle for Sparse Representations.

Suppose we have a nonzero vector $\mathbf{y} \in \mathbb{R}^n$ (a signal, say) and two orthobases $\mathbf{\Psi}$ and $\mathbf{\Phi}$. Then $\mathbf{y}$ can be represented either as a linear combination of columns of $\mathbf{\Psi}$ or as a linear combination of columns of $\mathbf{\Phi}$:

$$\mathbf{y} = \mathbf{\Psi}\alpha = \mathbf{\Phi}\beta.$$

Clearly, $\alpha$ and $\beta$ are uniquely defined. In a particularly important case, $\mathbf{\Psi}$ is simply the identity matrix and $\mathbf{\Phi}$ is the matrix of the Fourier transform. Then $\alpha$ is the time-domain representation of $\mathbf{y}$ while $\beta$ is the frequency-domain representation.

For certain pairs of bases $\mathbf{\Psi}$ $\mathbf{\Phi}$, an interesting phenomenon occurs: either $\alpha$ can be sparse, or $\beta$ can be sparse, but not both! In fact, we have the inequality [54, 49, 65]

$$(32) \qquad \text{(Uncertainty Principle 1)}: \quad ||\alpha||_0 + ||\beta||_0 \geq 2/\mu(\mathbf{A}).$$

So if the mutual coherence of two bases is small, then $\alpha$ and $\beta$ cannot both be very sparse. For example, if, as above, $\mathbf{\Psi}$ is the identity and $\mathbf{\Phi}$ is the Fourier matrix, then $\mu([\mathbf{\Psi}\ \mathbf{\Phi}]) = 1/\sqrt{n}$. It follows that a signal cannot have fewer than $\sqrt{n}$ nonzeros in both the time and frequency domains.

Heisenberg's classical uncertainty principle, in the discrete setting, says that if we view $\alpha$ and $\beta$ as probability distributions (by taking the absolute value of the entries and normalizing), then the product of variances $\sigma_\alpha^2 \sigma_\beta^2 \geq \text{const}$. In contrast, (32) gives a lower bound on the sum of the nonzeros. The uncertainty principle interpretation is developed at greater length in [54].

### 3.1.2. From Uncertainty to Uniqueness.

We now make a connection to the uniqueness problem. Consider the problem of finding a solution to $\mathbf{A}\mathbf{x} = [\mathbf{\Psi}\ \ \mathbf{\Phi}]\mathbf{x} = \mathbf{b}$ in light of the uncertainty principle (32). Suppose there are two solutions $\mathbf{x}_0$ and $\mathbf{x}_1$ for the underlying linear system, and that one is very sparse. We will see that the other one cannot also be very sparse. Necessarily, the difference $\mathbf{e} = \mathbf{x}_0 - \mathbf{x}_1$ must be in the null-space of $\mathbf{A}$. Partition $\mathbf{e}$ into subvectors $\mathbf{e}_\psi$ and $\mathbf{e}_\phi$ of the first $n$ entries and last $n$ entries of $\mathbf{e}$, respectively. We have

$$(33) \qquad \mathbf{\Psi}\mathbf{e}_\Psi = -\mathbf{\Phi}\mathbf{e}_\Phi = \mathbf{y} \neq 0.$$

The vector $\mathbf{y}$ is nonzero because $\mathbf{e}$ is nonzero and both $\mathbf{\Psi}$ and $\mathbf{\Phi}$ are nonsingular. Now invoke (32):

$$(34) \qquad \|\mathbf{e}\|_0 = \|\mathbf{e}_\Psi\|_0 + \|\mathbf{e}_\Phi\|_0 \geq \frac{2}{\mu(\mathbf{A})}.$$

Since $\mathbf{e} = \mathbf{x}_0 - \mathbf{x}_1$, we have

$$(35) \qquad \text{(Uncertainty Principle 2)}: \quad \|\mathbf{x}_0\|_0 + \|\mathbf{x}_1\|_0 \geq \|\mathbf{e}\|_0 \geq \frac{2}{\mu(\mathbf{A})}.$$

In other words, any two distinct solutions of the linear system $[\mathbf{\Psi}\ \mathbf{\Phi}]\mathbf{x} = \mathbf{b}$ cannot both be very sparse.

In fact, for the general matrix $\mathbf{A}$ we have also obtained such a rule in (5) as a byproduct of the proof for the uniqueness result. Although this inequality is posed in terms of the *spark* of $\mathbf{A}$, it can also be recast in terms of the mutual coherence, due to Lemma 4. Interestingly, the lower bound for the general case becomes $1 + 1/\mu(\mathbf{A})$. The general case bound is nearly a factor of 2 weaker than (35), because (35) uses the special structure $\mathbf{A} = [\mathbf{\Psi}\ \mathbf{\Phi}]$.

Returning to the case of a dictionary formed by concatenating two orthobases, $\mathbf{A} = [\mathbf{\Psi}\ \mathbf{\Phi}]$, a direct consequence of inequality (35) is a uniqueness result of the flavor already discussed in section 2.1: if a solution has fewer than $1/\mu(\mathbf{A})$ nonzeros, then any other solution must be "denser." Notice again that this specific case of the special structure of $\mathbf{A}$ yields a stronger uniqueness result than the one in Theorem 5.

### 3.1.3. Equivalence of Pursuit Algorithms.

Given the uniqueness of a sufficiently sparse solution of $[\mathbf{\Psi}\ \mathbf{\Phi}]\mathbf{x} = \mathbf{b}$, it becomes natural to ask how specific algorithms perform. A result, similar to Theorem 7, was obtained in [65, 73], showing that

$$\|\mathbf{x}\|_0 < \frac{\sqrt{2} - 0.5}{\mu(\mathbf{A})} \tag{36}$$

ensures that BP finds the proper (sparsest) solution. This was the first result of its kind; only later was the more general $\mathbf{A}$ case addressed. This result is better than the general result in Theorem 7 by a factor of almost 2. (A similar result holds for GAs in the two-orthobasis case; however, we are unaware of a citable publication.)

Intermediate between the two-orthobasis case and the general dictionary case is the case of concatenating $N$ orthogonal bases. Surprisingly, while concatenations of two orthobases can only have a coherence as small as $1/\sqrt{n}$, we can concatenate $N = n+1$ specially chosen orthobases together and still get coherence $1/\sqrt{n}$. So there are very large dictionaries with good coherence! This important result was first found by Emmanuel Knill in the theory of quantum error-correcting codes, where the special orthobases are called nice error bases and coherence is important for reasons unrelated to our interests here [11, 150]. The abovementioned uniqueness and equivalence theorems have been generalized to concatenations of several orthobases in [86, 47].

## 3.2. From Exact to Approximate Solution.

### 3.2.1. General Motivation.

The exact constraint $\mathbf{Ax} = \mathbf{b}$ is often relaxed, with approximate equality measured using the quadratic penalty function $Q(\mathbf{x}) = \|\mathbf{Ax} - \mathbf{b}\|_2^2$. Such relaxation allows us to (i) define a quasi-solution in case no exact solution exists; (ii) exploit ideas from optimization theory; and (iii) measure the quality of a candidate solution.

Following the rationale of the previous sections, one may reconsider $(P_0)$ and tolerate a slight discrepancy between $\mathbf{Ax}$ and $\mathbf{b}$. We define an error-tolerant version of $(P_0)$, with error tolerance $\delta > 0$, by

$$(P_0^\delta): \qquad \min_{\mathbf{x}} \ \|\mathbf{x}\|_0 \ \text{ subject to } \ \|\mathbf{b} - \mathbf{Ax}\|_2 \leq \delta. \tag{37}$$

The $\ell_2$ norm used here for evaluating the error $\mathbf{b} - \mathbf{Ax}$ can be replaced by other options, such as $\ell_1$, $\ell_\infty$, or a weighted $\ell_2$ norm.

In this problem a discrepancy of size $\delta$ is permitted between a proposed representation $\mathbf{Ax}$ and a signal $\mathbf{b}$. When $(P_0)$ and $(P_0^\delta)$ are applied on the same problem instance, the error-tolerant problem must always give results at least as sparse as those arising in $(P_0)$. Indeed, for a typical general problem instance $(\mathbf{A}, \mathbf{b})$, the solution of $(P_0)$ will have $n$ nonzeros. On the other hand, in some real-world problems (see below), although the solution of $(P_0)$ would have $n$ nonzeros, the solution of $(P_0^\delta)$ can be seen to have far fewer.

An alternative and more natural interpretation of the problem $(P_0^\delta)$ is one of noise removal. Consider a sufficiently sparse vector $\mathbf{x}_0$ and assume that $\mathbf{b} = \mathbf{Ax}_0 + \mathbf{z}$, where $\mathbf{z}$ is a nuisance vector of finite energy $\|\mathbf{z}\|_2^2 = \delta^2$. Roughly speaking $(P_0^\delta)$ aims to find $\mathbf{x}_0$, i.e., to do roughly the same thing as $(P_0)$ would do on noiseless data $\mathbf{b} = \mathbf{Ax}_0$.

Several papers study this problem [163, 48, 47, 157, 80], and we briefly discuss some of what is now known. Results are in some ways parallel to those in the noiseless case, although the notions of uniqueness and equivalence no longer apply—they are replaced by the notion of stability.

#### 3.2.2. Stability of the Sparsest Solution.

THEOREM 8 (stability of $(P_0^\delta)$ [48]). *Consider the instance of problem $(P_0^\delta)$ defined by the triplet $(\mathbf{A}, \mathbf{b}, \delta)$. Suppose that a sparse vector $\mathbf{x}_0 \in \mathbb{R}^m$ satisfies the sparsity constraint $\|\mathbf{x}_0\|_0 < (1 + 1/\mu(\mathbf{A}))/2$ and gives a representation of $\mathbf{b}$ to within error tolerance $\delta$ (i.e., $\|\mathbf{b} - \mathbf{A}\mathbf{x}_0\|_2 \le \delta$). Every solution $\mathbf{x}_0^\delta$ of $(P_0^\delta)$ must obey*

$$\|\mathbf{x}_0^\delta - \mathbf{x}_0\|_2^2 \le \frac{4\delta^2}{1 - \mu(\mathbf{A})(2\|\mathbf{x}_0\|_0 - 1)}. \tag{38}$$

This result parallels Theorem 5, to which it reduces for the case of $\delta = 0$. A result of similar flavor, proposing a simple and constructive test for near-optimality of the solution of $(P_0^\delta)$, appears in [85].

**3.2.3. Pursuit Algorithms.** Since $(P_0)$ is impractical to solve in the general case, it seems unlikely that a direct attack on $(P_0^\delta)$ is a sensible goal. The pursuit algorithms discussed above can be adapted to allow error tolerances; how will they perform? Referring to the two options we had for devising such algorithms—the greedy approach and the regularization of the $\ell_0$ functional—variants of these methods may be investigated. Consider, for example, the GA described in Exhibit 1—OMP. By choosing $\epsilon_0 = \delta$ in the stopping rule, the algorithm accumulates nonzero elements in the solution vector until the constraint $\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2 \le \delta$ is satisfied.

Similarly, relaxing $\ell_0$ to an $\ell_1$ norm, we get the following variant of $(P_1)$, known in the literature as *basis pursuit denoising* (BPDN) [24]:

$$(P_1^\delta): \qquad \min_{\mathbf{x}} \; \|\mathbf{W}\mathbf{x}\|_1 \;\text{ subject to }\; \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2 \le \delta, \tag{39}$$

where $\mathbf{W}$ is again a diagonal positive-definite weight matrix. This can be written as a standard problem in linear optimization under quadratic and linear inequality constraints. Such problems are very well studied by specialists in optimization and there are many practical methods for solving them—practically the whole of modern convex optimization theory is applicable, particularly the recent advances in solving large systems by interior-point and related methods [24]. We cannot begin to review that literature here, and instead discuss a very simple approach, suitable for readers without background in convex optimization.

For an appropriate Lagrange multiplier $\lambda$, the solution to (39) is precisely the solution to the unconstrained optimization problem

$$(Q_1^\lambda): \qquad \min_{\mathbf{x}} \; \lambda\|\mathbf{W}\mathbf{x}\|_1 + \frac{1}{2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2, \tag{40}$$

where the Lagrange multiplier $\lambda$ is a function of $\mathbf{A}$, $\mathbf{b}$, and $\delta$.

General methods as discussed above are a good way to get a reliable solution to $(P_1^\delta)$ with little programming effort—one only has to set up $(P_1^\delta)$ as a problem of the type the optimizer can solve. However, for large-scale applications, general purpose optimizers seem slow and can perhaps be improved by special purpose techniques. We mention three.

**Iteratively Reweighted Least Squares.** A simple strategy to attack $(Q_1^\lambda)$ is the iteratively reweighted least squares (IRLS) algorithm [97, 139, 138]. Setting $\mathbf{X} = \mathrm{diag}(|\mathbf{x}|)$, we have $\|\mathbf{x}\|_1 \equiv \mathbf{x}^T\mathbf{X}^{-1}\mathbf{x}$. Thus we may view the $\ell_1$ norm as an (adaptively weighted) version of the squared $\ell_2$ norm. Given a current approximate solution $\mathbf{x}_{k-1}$, set $\mathbf{X}_{k-1} = \mathrm{diag}(|\mathbf{x}_{k-1}|)$ and attempt to solve

$$(M_k): \qquad \min_{\mathbf{x}} \quad \lambda \mathbf{x}^T \mathbf{W} \mathbf{X}_{k-1}^{-1} \mathbf{x} + \frac{1}{2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2; \tag{41}$$

this is a quadratic optimization problem, solvable using standard linear algebra. Obtain an (approximate) solution $\mathbf{x}_k$ (say); a diagonal matrix $\mathbf{X}_k$ is constructed with the entries of $\mathbf{x}_k$ on the diagonal, and a new iteration can begin. This algorithm is formally described in Exhibit 2.

**Task:** Find $\mathbf{x}$ that approximately solves $(Q_1^\lambda)$: $\min_{\mathbf{x}} \ \ \lambda\|\mathbf{W}\mathbf{x}\|_1 + \frac{1}{2} \cdot \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2$.

**Initialization:** Initialize $k = 0$, and set

- The initial approximation $\mathbf{x}_0 = \mathbf{1}$.
- The initial weight matrix $\mathbf{X}_0 = \mathbf{I}$.

**Main Iteration:** Increment $k$ by $1$, and apply these steps:

- **Regularized Least Squares:** Approximately solve the linear system

  $$\left(2\lambda \mathbf{W}\mathbf{X}_{k-1}^{-1} + \mathbf{A}^T\mathbf{A}\right)\mathbf{x} = \mathbf{A}^T\mathbf{b}$$

  iteratively (several conjugate gradient iterations may suffice), producing result $\mathbf{x}_k$.
- **Weight Update:** Update the diagonal weight matrix $\mathbf{X}$ using $\mathbf{x}_k$: $X_k(j,j) = |x_k(j)| + \epsilon$.
- **Stopping Rule:** If $\|\mathbf{x}_k - \mathbf{x}_{k-1}\|_2$ is smaller than some predetermined threshold, stop. Otherwise, apply another iteration.

**Output:** The desired result is $\mathbf{x}_k$.

**Exhibit 2.** *The IRLS strategy for approximately solving* $(Q_1^\lambda)$.

**Iterative Thresholding.** IRLS loses much of its appeal when facing the very large-scale problems which are of most interest in today's applications; see below. An alternative family of iterative approximate solution techniques was developed in [76, 75, 32, 62, 5, 67] and coined *iterated shrinkage algorithms*; these methods are very easy to implement, and in some sense very intuitive to apply.

We first remark that if $\mathbf{A}$ is a square unitary matrix, the problem $(Q_1^\lambda)$ has a simple, noniterative closed-form solution, $\mathbf{x}_1^\lambda$, say; it can be found as follows. First, apply $\mathbf{A}^T$ to the vector $\mathbf{b}$, obtaining a preliminary solution $\tilde{\mathbf{x}}$, which, in favorable cases, exhibits a few large entries rising above many small "noise" entries—like daisies sticking up above the weeds. Second, apply soft thresholding, setting to zero the entries below the threshold and shrinking the other entries toward zero. This is done formally by defining the scalar function $\eta(x;\lambda) = \mathrm{sign}(x) \cdot (|x| - \lambda)_+$, inducing the vector function

$$\hat{\mathbf{x}} = \mathrm{Shrink}(\tilde{\mathbf{x}}; \lambda) \tag{42}$$

by elementwise application of $\eta : \hat{x}(j) = \eta(\tilde{x}(j); \lambda)$. Equation (42) is known as the "shrinkage" operation, since it clearly tends to shrink the magnitude of the entries of $\tilde{\mathbf{x}}$, while setting the small ones to zero [50, 51, 27, 43, 53, 52, 144, 124, 96].

In the general case where $\mathbf{A}$ is not a unitary matrix, we can apply this idea iteratively. A step toward such a generalization was proposed initially by Sardy, Bruce, and Tseng—their block-coordinate relaxation (BCR) algorithm considers matrices $\mathbf{A}$

that are concatenations of unitary matrices, and iteratively updates the solution one part at a time, using shrinkage [143]. Handling of the general case is somewhat more involved, as can be seen in [76, 75, 32, 62, 5, 67, 59]. Exhibit 3 spells out the details. The algorithm is particularly useful in large-scale problems, where $\mathbf{A}$ is defined not by an explicitly given matrix, but instead by an operator which we know how to apply rapidly. Note that in very large problems, OMP is not really practical, as it requires direct manipulation of the columns of $\mathbf{A}$.

**Task:** Find $\mathbf{x}$ that approximately solves $(Q_1^\lambda)$: $\min_{\mathbf{x}} \quad \lambda\|\mathbf{W}\mathbf{x}\|_1 + \frac{1}{2}\cdot\|\mathbf{b}-\mathbf{A}\mathbf{x}\|_2^2$.

**Initialization:** Initialize $k=0$, and set

- The initial solution $\mathbf{x}^0 = 0$.
- The initial residual $\mathbf{r}^0 = \mathbf{b} - \mathbf{A}\mathbf{x}^k = \mathbf{b}$.
- Compute $\mathbf{W}$ and normalize $\mathbf{A}$, replacing it with $\mathbf{A}\mathbf{W}^{-1}$.

**Main Iteration:** Increment $k$ by $1$, and apply these steps:

- **Back-Projection:** Compute $\mathbf{e} = \mathbf{A}^T\mathbf{r}^{k-1}$ and multiply by $\mathbf{w}$ entrywise.
- **Shrinkage:** Compute $\mathbf{e}_s = \text{Shrink}\left(\mathbf{x}^{k-1} + \mathbf{e}\right)$ with threshold $\lambda$.
- **Line Search:** Choose $\mu$ to minimize the real-valued function $J(\mathbf{x}^{k-1} + \mu(\mathbf{e}_s - \mathbf{x}^{k-1}))$, where $J$ is the objective function of $(Q_1^\lambda)$.
- **Update Solution:** Compute $\mathbf{x}^k = \mathbf{x}^{k-1} + \mu(\mathbf{e}_s - \mathbf{x}^{k-1})$ .
- **Update Residual:** Compute $\mathbf{r}^k = \mathbf{b} - \mathbf{A}\mathbf{x}^k$.
- **Stopping Rule:** If $\|\mathbf{x}^k - \mathbf{x}^{k-1}\|_2^2$ is smaller than some predetermined threshold, stop. Otherwise, apply another iteration.

**Output:** The result is $\mathbf{W}^{-1}\mathbf{x}^k$.

**Exhibit 3.** *An iterated shrinkage algorithm for solving* $(Q_1^\lambda)$.

**Stepwise Algorithms: LARS and Homotopy.** Certain heuristic methods inspired by true $(P_1)$ solvers are currently attracting serious interest; these are not entirely greedy—they can be viewed as following some but not all of the principles of a true $\ell_1$ solver. These include LARS [60] and polytope faces pursuit [133, 134, 135]. In practice these can work extremely well [160], for example, from the viewpoint of phase transition, as discussed in section 3.3.3.

It is actually possible to solve $(P_1^\delta)$ by an algorithm very reminiscent of OMP. Suppose the matrix $\mathbf{A}$ has normalized columns $\|\mathbf{a}_i\|_2 = 1$, $i = 1, \ldots, m$. Starting from $\mathbf{x}^0 = 0$ and support set $\mathcal{S}_0 = \emptyset$, proceed stepwise. At the $k$th step, find an index $i$ of a current zero, $\mathbf{x}^{k-1}(i) = 0$, for which the corresponding column of $\mathbf{A}$ makes the highest correlation with the current residual among all such columns:

$$\max_{i \notin \mathcal{S}_{k-1}} |\langle \mathbf{y} - \mathbf{A}\mathbf{x}^{k-1}, \mathbf{a}_i \rangle|.$$

Label that entry $i_k$ and form the new support set $\mathcal{S}_k = \mathcal{S}_{k-1} \cup \{i_k\}$. Now obtain $\mathbf{x}^k$ with nonzeros at positions in $\mathcal{S}_k$. So far this sounds the same as OMP. However, we do not solve for $\mathbf{x}^k$ by least squares. Instead, choose the nonzeros in $\mathbf{x}^k$ so that the residual $\mathbf{y} - \mathbf{A}\mathbf{x}^k$ is equicorrelated with every column $\mathbf{a}_i$, $i \in \mathcal{S}_k$:

$$|\langle \mathbf{y} - \mathbf{A}\mathbf{x}^k, \mathbf{a}_i \rangle| = \text{const} \qquad \forall i \in \mathcal{S}_k.$$

OMP would instead demand that the residual be uncorrelated. This is the LARS algorithm of Efron et al. [60].

Yaakov Tsaig has demonstrated an analogue of Theorems 6 and 7 for LARS: under incoherence and sufficient sparsity, LARS takes $\|\mathbf{x}_0\|_0$ steps and stops, having produced the unique sparsest solution [160]. So LARS is in some sense equally as good as OMP and BP in this setting. In fact, more is true. Now modify the LARS algorithm so that at each stage either a new term can enter or an old term can leave the support, seeking to maintain that for each $i \in \mathcal{S}_k$,

$$|\langle \mathbf{y} - \mathbf{A}\mathbf{x}^k, \mathbf{a}_i \rangle| = \text{const} > \max_{j \notin \mathcal{S}_k} |\langle \mathbf{y} - \mathbf{A}\mathbf{x}^k, \mathbf{a}_j \rangle|.$$

With this variation, we have the LARS-LASSO algorithm, also known as the homotopy algorithm of Osborne, Presnell, and Turlach [129]. This new algorithm solves $(P_1)$: continuing it until the residual is zero gives the solution to the $\ell_1$ minimization problem. Tsaig has also shown that under incoherence and sufficient sparsity, LARS-LASSO/homotopy takes $\|\mathbf{x}_0\|_0$ steps and stops, having produced the unique sparsest solution [160]. The results at each step are the same as LARS. In short, a small modification of OMP produces a stepwise algorithm that exactly solves $(P_1)$. This helps explain the similarity of the coherence-based results for the two methods.

A great deal of algorithmic progress was made while this paper was in review and revision. We mention only two examples. Candès and Romberg [14] have developed a fast approximate $\ell^1$ solver using projections onto convex sets. Stephen Boyd and coworkers [100] have found a way to speed up standard interior-point methods so that, when the solution is sparse, they run quickly.

**3.2.4. Performance of Pursuit Algorithms.** Can pursuit methods approximately solve $(P_0^\delta)$? We quote two theorems from [48]; the first corresponds to BPDN, and the second to OMP. In the first, stability of the solution is guaranteed when a sufficiently sparse solution exists. While similar in flavor to the stability of $(P_0^\delta)$ claimed in Theorem 8, it is weaker in two ways—the sparsity requirement is more strict, and the tolerated error level is larger.

THEOREM 9 (stability of BPDN [48]). *Consider the instance of problem $(P_1^\delta)$ defined by the triplet $(\mathbf{A}, \mathbf{b}, \delta)$. Suppose that a vector $\mathbf{x}_0 \in \mathbb{R}^m$ satisfies the sparsity constraint $\|\mathbf{x}_0\|_0 < (1 + 1/\mu(\mathbf{A}))/4$ and gives a representation of $\mathbf{b}$ to within error tolerance $\delta$; $\|\mathbf{b} - \mathbf{A}\mathbf{x}_0\|_2 \le \delta$. The solution $\mathbf{x}_1^\delta$ of $(P_1^\delta)$ must obey*

$$\|\mathbf{x}_1^\delta - \mathbf{x}_0\|_2^2 \le \frac{4\delta^2}{1 - \mu(\mathbf{A})(4\|\mathbf{x}_0\|_0 - 1)}. \tag{43}$$

In addition to stability, results on successful recovery of the support would also be of interest. The work reported in [48, 80, 157] offers such results, but requires more strict (and thus less realistic) conditions. Similarly, considering the above as a signal denoising procedure, there is interest in the expected performance. The work reported in [77, 78] uses an information theoretic point of view to provide such analysis for the extreme (very weak and very strong) noise cases.

Turning to OMP, the following theorem taken from [48] establishes both stability and correct recovery of the support. Notice, however, that for these to hold true, the magnitude of the smallest nonzero entry in the "ideal" solution $\mathbf{x}_0$ must be sufficiently large compared to the "noise level" ($\delta$). Results of the same flavor are derived in [82, 159, 157].

THEOREM 10 (performance of OMP [48, 159, 157]). *Consider the OMP algorithm applied to the problem instance $(P_1^\delta)$ with the triplet $(\mathbf{A}, \mathbf{b}, \delta)$. Suppose that a vector*

$\mathbf{x}_0 \in \mathbb{R}^m$ *satisfies the sparsity constraint*

$$\|\mathbf{x}_0\|_0 < \frac{1}{2}\left(1+\frac{1}{\mu(\mathbf{A})}\right) - \frac{\delta}{\mu(\mathbf{A})\cdot x_{min}}, \tag{44}$$

*where $x_{min}$ is the smallest nonzero entry (in absolute value) in $\mathbf{x}_0$. Assume further that $\mathbf{x}_0$ gives a representation of $\mathbf{b}$ to within error tolerance $\delta$ (i.e., $\|\mathbf{b}-\mathbf{A}\mathbf{x}_0\|_2 \le \delta$). The result produced by OMP must obey*

$$\|\mathbf{x}_{OMP} - \mathbf{x}_0\|_2^2 \le \frac{\delta^2}{1-\mu(\mathbf{A})(\|\mathbf{x}_0\|_0 - 1)}. \tag{45}$$

*Furthermore, OMP is guaranteed to recover a solution with the correct support.*

There are various ways in which these results are far weaker than one would like. However, they show that adopting sparsity as a goal can lead to sensible results, stable under additive noise.

### 3.3. Beyond Coherence Arguments.

**3.3. Beyond Coherence Arguments.** The analysis presented so far—largely based on coherence arguments—presents a simple but limited portrait of the ability of concrete algorithms to find sparse solutions and near-solutions. We now briefly point to the interesting and challenging research territory that lies beyond coherence. We start with some simple simulations.

**3.3.1. Empirical Evidence.** Consider a random matrix $\mathbf{A}$ of size $100 \times 200$, with entries independently drawn at random from a Gaussian distribution of zero mean and unit variance, $\mathcal{N}(0,1)$. The *spark* of this matrix is 101 with probability 1, implying that every solution for the system $\mathbf{A}\mathbf{x} = \mathbf{b}$ with less than 51 entries is necessarily the sparsest one possible, and, as such, it is the solution of $(P_0)$. By randomly generating such sufficiently sparse vectors $\mathbf{x}$ (choosing the nonzero locations uniformly over the support in random and their values from $\mathcal{N}(0,1)$), we generate vectors $\mathbf{b}$. This way, we know the sparsest solution to $\mathbf{A}\mathbf{x} = \mathbf{b}$, and we shall be able to compare this to algorithmic results.

The graph presented in Figure 2 shows the success rate for both OMP and the BP in recovering the true (sparsest) solution. For each cardinality, 100 repetitions were conducted and their results averaged. The value of the mutual coherence of $\mathbf{A}$ in this experiment is $\mu(\mathbf{A}) = 0.424$, so that only for cardinalities lower than $(1 + 1/\mu(\mathbf{A}))/2 = 1.65$ are pursuit methods guaranteed to succeed. As we can see, both pursuit algorithms succeed in the recovery of the sparsest solution for $1 \le \|\mathbf{x}\|_0 \le 26$, far beyond the coverage of Theorems 6 and 7. We can also see that the GA (OMP) is performing somewhat better than the BP.

The graph presented in Figure 3 is similar, showing the success rate for OMP and BP for the approximated case (solution of $(P_0^\delta)$). We generate a random vector $\mathbf{x}_0$ with a prespecified cardinality of nonzeros. It is normalized so $\|\mathbf{A}\mathbf{x}_0\|_2 = 1$. We compute $\mathbf{b} = \mathbf{A}\mathbf{x}_0 + \mathbf{z}$, where $\mathbf{z}$ is a random vector with predetermined norm $\|\mathbf{z}\|_2 = \delta = 0.1$. Thus, the original vector $\mathbf{x}_0$ is a feasible solution for $(P_0^\delta)$ and close to the optimal due to its sparsity. Given an approximate solution $\mathbf{x}$ (by IRLS, with a line search determining $\lambda$ so that the desired misfit $Q(\mathbf{x})$ is obtained), the stability of the process is tested by $\|\mathbf{x} - \mathbf{x}_0\|_2 \le \delta$. As can be seen, both methods do very well for $1 \le \|\mathbf{x}_0\|_0 \le 15$, while the stability results set much lower expectation bounds for stability to hold.

**3.3.2. Formal Machinery: Suites and Ensembles.** Such simulations show that coherence does not tell the whole story. When working with matrices having spe-

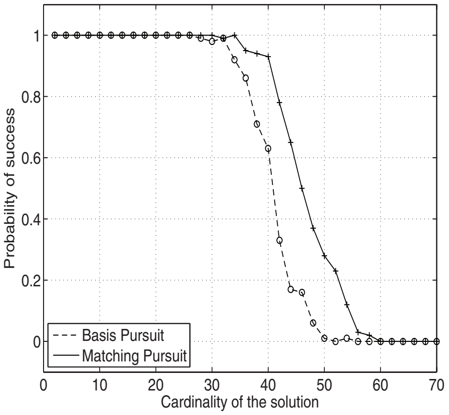


**Fig. 2** *Probability of success of pursuit algorithms in the recovery of the sparsest solution of the linear system $\mathbf{Ax} = \mathbf{b}$. The results are shown as a function of the cardinality of the desired solution.*

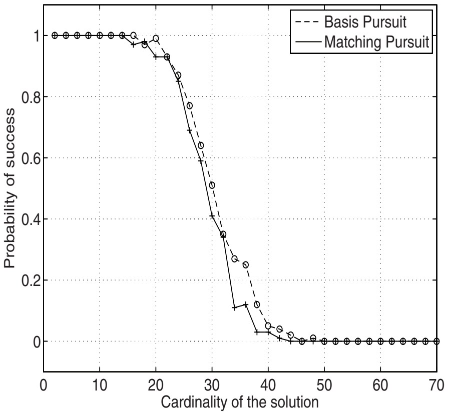


**Fig. 3** *Probability of success of pursuit algorithms in the stable recovery of the sparsest solution of the linear system $\mathbf{Ax} = \mathbf{b}$ in the presence of noise.*

cific properties, or with problems exhibiting structured sparsity, we might find that coherence gives very weak guarantees compared to what actually happens.

Following [161], we introduce a notion to bring information about properties of the matrix $\mathbf{A}$ and the sparsest solution $\mathbf{x}_0$ formally into the picture. A *problem suite* $\mathcal{S} = \mathcal{S}(\mathcal{A}, \mathcal{X})$ is defined by an ensemble of matrices $\mathcal{A}$ of a given shape and a collection of solution vectors $\mathcal{X}$ obeying some sparsity condition. Here are some examples of matrix ensembles which can be used to form problem suites:

- The incoherent ensemble $\mathcal{A}_{IE}(\mu; n, m)$, consisting of all $n \times m$ matrices with normalized columns and $\mu(A) \leq \mu$.
- The Gaussian ensemble $\mathcal{A}_{GE}(n, m)$, consisting of all $n \times m$ with entries drawn from a Gaussian i.i.d. $N(0, 1/n)$ distribution.
- The partial Fourier ensemble, consisting of all $n \times m$ matrices with $n$ rows drawn at random without replacement from the $m \times m$ Fourier matrix.
- The time-frequency dictionary—a singleton ensemble $\mathcal{A}_{\mathcal{TF}}$, consisting of just $\mathbf{A} = [\mathbf{I}\ \mathbf{F}]$, where $\mathbf{F}$ is the $n \times n$ Fourier matrix.

Here are some collections of solution vectors which can be used to form problem suites:

- The $k$-sparse collection $L_0(k)$, consisting of all vectors $\mathbf{x} \in \mathbb{R}^m$ with at most $k$ nonzero entries.
- The Bernoulli–Gaussian ensemble $BG(k)$, consisting of random vectors in $\mathbb{R}^m$ with sites of nonzeros chosen at random by tossing a coin with probability $\epsilon = k/m$, and with the nonzero sites having values chosen from a standard Gaussian distribution.

So far we have mostly discussed the incoherent problem suite $\mathcal{S}(\mathcal{A}_{IE}(\mu; n, m), L_0(k))$ consisting of the incoherent matrix ensemble and the $k$-sparse collection. However, in the previous subsection we were implicitly considering the Gaussian problem suite $\mathcal{S}(\mathcal{A}_{GE}(n, m), BG(k))$. These two suites are only two among many, but they do represent extremes of a kind. For the incoherent suite it is natural to study the *worst-case* behavior of algorithms, while for the Gaussian suite it is natural to study the *typical behavior*. The last subsection's simulations show that the worst-case behavior over the incoherent suite can be very different from the typical behavior over the Gaussian suite.

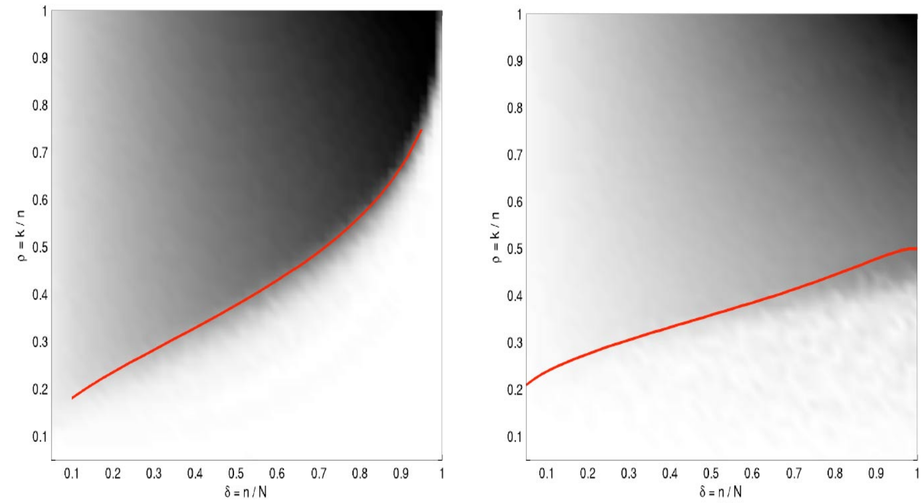
In fact, this distinction between worst-case and typical behavior has been known for some time. In [54], it was shown empirically that for the time-frequency dictionary and the typical $k$-sparse sequence, something dramatically stronger than the uncertainty principle was true; while the uncertainty principle guarantees that the number of nonzeros in the combined time-frequency analysis must exceed $\sqrt{n}$; in fact, the typical number is closer to $n$. Also, in [49] simulations very much like those reported above in section 3.3.1 were presented to show that the equivalence between $\ell_1$ and $\ell_0$ representations is typical at surprisingly weak levels of sparsity; in fact, for the time-frequency dictionary, while the coherence theory would be able to guarantee equivalence only when there are fewer than $\sqrt{n}$ nonzeros in the solution, equivalence was found to be typical when there are fewer than about $n/4$ nonzeros.

**3.3.3. Phase Transitions in Typical Behavior.** Simulation studies of typical-case behavior of algorithms exhibit surprising regularities. Consider the problem suite $\mathcal{S}(\mathcal{A}_{GE}(n, m), L_0(k))$ and define variables $\delta = n/m$ and $\rho = k/n$. In the interesting case $m > n$, $\delta \in (0, 1)$, while $k < n$, so that $\rho \in (0, 1)$ as well. The simulations in section 3.3.1 explored the case $\delta = .5$ and $0 < \rho < .7$; Figure 2 revealed a relatively rapid drop in the probability of successful recovery by BP and OMP as $\rho$ increased from .3 to .6.

Such phenomena have been observed for a variety of sparsity-seeking algorithms. A typical example is given in Figure 4. Panel (a), taken from [57], depicts the unit square of interesting $\delta - \rho$ behavior; the shaded attribute displays simulation results for the probability that the solutions to $\ell_1$ and $\ell_0$ are equivalent. Just as in Figure 2, there is a relatively rapid transition from probability near one to probability near zero as $\rho$ increases. Panel (b) is taken from [161] and depicts the behavior of an iterative thresholding algorithm StOMP (stagewise OMP). The shaded attribute displays the fraction of truly nonzero coefficients recovered by StOMP; again there is a rapid transition from nearly 100% success to nearly 0% success.

As the problem size increases, the transition from typicality of success to typicality of failure becomes increasingly sharp—in the large-$n$ limit, perfectly sharp. A rigorous result from [44, 56, 57] explains the meaning of the curve in panel (a).

THEOREM 11. *Fix a $(\delta, \rho)$ pair. At problem size $n$, set $m_n = \lfloor n/\delta \rfloor$ and $k_n = \lfloor n\rho \rfloor$. Draw a problem instance $\mathbf{y} = \mathbf{A}\mathbf{x}$ at random with $\mathbf{A}$ an $n \times m_n$ matrix from the Gaussian ensemble and $\mathbf{x}$ a vector from the $k$-sparse collection $L_0(k_n)$.*



(a) (b)

**Fig. 4** (a) *Phase transition behavior of $\ell_1$ minimization. Shaded attribute: fraction of cases in which $\ell_1$ minimization successfully finds the sparsest solution (in the range* 0*-black to* 1*-white). The curve displays the function $\rho_W$ defined in Theorem* 12*. The curve closely follows the rapid change in the shaded attribute.* (b) *Phase transition behavior of StOMP. Shaded attribute: fraction of cases in which StOMP successfully finds the sparsest solution. The red curve displays the function $\rho_{StOMP}$ defined in* [59]*. The curve closely follows the rapid change in the shaded attribute. Both panels use the coordinates $\delta = n/m$ (ratio of number of equations to number of unknowns) and $\rho = k/n$ (ratio of number of nonzeros to number of equations). The experiments explore a grid of $\rho - \delta$ values at problem size $m =$ 1600. The underlying matrix ensemble is Gaussian.*

*There is a function $\rho_W(\delta) > 0$ with the following property: As $n$ increases, the probability that, for such a random problem instance, the two problems $(P_1)$ and $(P_0)$ have the same solution tends to zero if $\rho > \rho_W(\delta)$ and tends to* 1 *if $\rho < \rho_W(\delta)$.*

In other words, for large $n$, there are really two "phases" in the $(\delta, \rho)$ "phase plane," a success phase and a failure phase. The curve $\rho(\delta)$ has an interesting form for small $\delta > 0$:

Theorem 12 (see [57]).

$$\rho_W(\delta) = \frac{1}{2 \log 1/\delta}(1 + o(1)), \qquad \delta \to 0.$$

Informally, in the setting of Gaussian random matrices and $m \gg n$, we have a threshold

$$\frac{n}{2 \log(m/n)}.$$

If $k$ is a bit larger than this threshold, we are in the failure phase, while if $k$ is a bit smaller, we are in the success phase. In other words, *if the number $n$ of Gaussian "measurements" of a $k$-sparse vector exceeds $2 \log(m/n)k$, the vector is highly likely to be recovered by $\ell_1$ minimization.* The techniques underlying the proofs combine exact identities from convex integral geometry with asymptotic analysis.

The curve in panel (b), concerning the iterative thresholding scheme StOMP, has a similar large-$n$ interpretation; there is a rigorous result establishing the existence of a curve $\rho_{STOMP}(\delta)$ separating the phases and a rigorous result showing how to compute that curve [59].

As an empirical matter, the existence of phase transitions is well established for several algorithms, particularly including OMP. In fact, as Figure 2 showed, OMP can exhibit phase transition performance competitive to BP. However, to the best of our knowledge, there is at the moment no theoretical calculation giving a curve $\rho_{OMP}$ which matches the empirically observed behavior. Part of the problem may be that unlike the case with BP, the phase transition for OMP is sensitive to the distribution of the nonzero elements in the sparsest solution. The empirical phase transition is different (lower) for OMP when the coefficients are random signs $\pm 1$ than when they are random Gaussian! In particular, the apparent advantage of OMP over BP seen in Figure 2 disappears if we replace Gaussian-distributed nonzeros in $x$ by random $\pm 1$ terms.

Over the last two years a wide range of rigorous mathematical analysis has been published to address the large-$n$ setting we have just discussed. It covers several sparsity-seeking algorithms and a variety of assumptions about the matrix ensemble and the sparse solution. It is difficult here to summarize all this work in a short space; we limit ourselves to a few examples.

- Candès, Romberg, and Tao had a great impact in 2004 when announcing that they could prove typicality of equivalence of $(P_1)$ and $(P_0)$, where the sparsity control parameter $k$ could be as large as $n/\log(n)^6$ and the matrix was drawn from the partial Fourier ensemble [15]. Here the $1/\log(n)^6$ factor is unnecessarily small. Donoho effectively showed that for matrices from the Gaussian ensemble and $n/m \sim \delta$, equivalence could hold with sparsity control parameter $k \approx r(\delta)n$ for an unspecified function $r > 0$. Candès and Tao considered random Gaussian matrices and were able to show that $k \leq r(m/n)n$ was sufficient for equivalence for a certain explicit function $r$ [18]. These qualitative results opened the way to asking for the precise quantitative behavior, i.e., for $\rho_W$ above.
- Tropp, Gilbert, and coworkers [158] studied running OMP over the problem suite consisting of the Gaussian matrix ensemble and $k$-sparse coefficients solution; they showed that the sparsest solution is found with high probability provided $k \leq c \cdot n/\log(n)$. Since the empirical evidence suggests that the true state of affairs is a phase transition at $k \approx \rho_{OMP}(m/n)n$ for some function $\rho_{OMP}$, this important result is still somewhat weaker than what we expect to be the case.

In general, many researchers making progress in asymptotic studies exploit ideas from geometric functional analysis and techniques from random matrix theory. Useful results from that literature include Szarek's bound on the singular values of a random Gaussian matrix [151, 152], which allows one to easily control the maximal and minimal singular values across all $n \times k$ submatrices; this principle has been used in [44, 18] in studying the equivalence of $(P_1)$ and $(P_0)$ and in [158] in studying OMP. Other fundamental ideas include Kashin's results on $n$-widths of the octahedron [98], Milman's quotient of a subspace theorem, and Szarek's volume bounds [132], all reflecting the miraculous properties of $\ell_1$ norms when restricted to random subspaces, which lie at the heart of the $(P_0)$–$(P_1)$ equivalence. Rudelson and Vershynin [140] have made very effective use of such geometric functional analysis tools in giving the shortest and simplest proofs that $k \leq r(m/n)n$ is sufficient for $\ell_1$–$\ell_0$ equivalence. It seems that these tools do not allow us to precisely pin down the location of the actual phase transitions. However, they are very flexible and widely applicable. Mendelson, Pajor, and Tomczak-Jagerman [119, 120] have been able to use geometric functional analysis techniques to establish $(P_0)$–$(P_1)$ equivalence for matrices with random $\pm 1$ entries. Such tools have also been used [45, 17] to get rigorous results on stability of solutions $(P_1^\epsilon)$, showing stability is obtained for $k \le r(m/n)n$.

At this point, the literature is growing so rapidly that it is difficult to do any justice at all to this field, its achievements, and its results. We will mention one particularly elegant result of Candès and Tao [18], which develops a tool going beyond coherence, now outranking coherence as the focus of attention in current research on sparse representation.

Definition 13. *An $n \times m$ matrix $\mathbf{A}$ is said to have the restricted isometry property $RIP(\delta; k)$ if each submatrix $\mathbf{A}_I$ formed by combining at most $k$ columns of $\mathbf{A}$ has its nonzero singular values bounded above by $1 + \delta$ and below by $1 - \delta$.*

Candès and Tao have shown that $\mathbf{A} \in RIP(.41; 2k)$ implies that $(P_1)$ and $(P_0)$ have identical solutions on all $k$-sparse vectors and, moreover, that $(P_1^\epsilon)$ stably approximates the sparsest near-solution of $\mathbf{y} = \mathbf{Ax} + \mathbf{z}$—with a reasonable stability coefficient.

The restricted isometry property is useful because it can be established using probabilistic methods. Thus, with high probability, a matrix $\mathbf{A}$ with Gaussian i.i.d. entries has this property for $k < c(1 + \log(m/n))n$ for some small positive constant $c > 0$. This can be shown, e.g., using Szarek's results on singular values of Gaussian matrices as in [44]. (Analysis of other matrices—for example, with $\pm 1$ entries—can be more challenging.) The restricted isometry property approach thus gives the qualitative bound $k < r(m/n)n$, which, as for other geometric functional analysis techniques, is qualitatively correct and very useful, but apparently smaller than the actual behavior of the phase transitions.

### 3.4. The Sparsest Solution of $\mathbf{Ax} = \mathbf{b}$: A Summary.

This section, like the previous ones, discusses a wealth of recent results on the study of the underdetermined linear system $\mathbf{Ax} = \mathbf{b}$ and the quest for its sparsest solution. Questions such as solvability of such problems, uniqueness of the sparsest solutions, extensions to approximate solutions, and so on, have been addressed over the past few years. Much work remains to be done, and we list several open questions and research directions.

- If $\mathbf{A}$ has some special structure, this structure can be exploited in order to obtain stronger uniqueness and equivalence claims. Such is indeed the case with concatenations of unitary matrices. Further work is required for other structured matrices, such as those whose columns are redundant wavelet bases, Gabor bases, and others. Of particular interest is the exploitation of the multiscale structure underlying these dictionaries. This was explicitly done in [49], but it seems that much more should be possible.
- As mentioned above, and quite surprisingly, a study of the performance of the GAs for the concatenation of two (or more) unitary matrices is missing. While this case has been studied thoroughly for BP, we have not seen a similar analysis for OMP or other greedy techniques.
- There is a need for fast algorithms for BP (both in the accurate and the approximate versions) that compete favorably with greedy methods. Perhaps a more ambitious goal would be an attempt to unify those methods and show a common ground for all of them. In this respect, the recent progress made on iterated shrinkage methods is one such promising research direction. The recent progress reported in section 3.2.3 (iterated shrinkage, LARS/LASSO, fast general solvers) indicates a great deal of ongoing work, so we can expect further progress in the near future.

- Still considering average performance, most existing results are of limited scope due to their asymptotic nature, or rather limiting assumptions (such as the structured matrix $\mathbf{A}$ in [16] or its random content in [44]). Derivation of stronger results that refer to specific matrices and bypass the use of the mutual coherence should be attempted. Indeed, the mutual coherence is a definition that stems from a worst-case point of view, and as such should be avoided.
- The uniqueness and equivalence claims we have shown so far are general and hold true uniformly for all $\mathbf{b}$. Can signal-dependent or representation-dependent theorems be derived, yielding stronger guarantees for uniqueness and equivalence?
- The mutual coherence suggests a way for bounding the *spark* by testing pairs of atoms from $\mathbf{A}$. Similar and better treatment may be possible by considering the behavior of triplets or k-sets of atoms for $k = 4, 5, \dots$.

We now turn to practical applications.

## 4. Sparsity-Seeking Methods in Signal Processing.

We now see that the problem of finding sparse representations of signal vectors can be given a meaningful definition and, contrary to expectation, can also be computationally tractable. These suggest that *sparsity-driven signal processing* is a valid research agenda and a potentially useful practical tool. We now develop this idea in more detail.

### 4.1. Priors and Transforms for Signals.

Consider a family of signals $\mathbf{y} \in \mathbb{R}^n$. To make our discussion more concrete, we assume, here and below, that each such signal is a $\sqrt{n} \times \sqrt{n}$ pixel image patch, representing natural and typical image content. Our discussion applies, with obvious changes, to other signal types: sound signals, seismic data, medical signals, financial information, and so on.

While image patches are scattered about in $\mathbb{R}^n$, they do not populate it uniformly. For example, we know that spatially smooth patches are frequently seen in images, whereas highly nonsmooth image content is rare. Talk of "typical behavior" would suggest the Bayesian approach to signal processing. In that approach, the researcher would model the probability density function (PDF) of images using a specific prior distribution $p(\mathbf{y})$ and then derive Bayesian algorithms under that specific assumption.

For example, consider the denoising problem, where we observe a noisy version $\tilde{\mathbf{y}} = \mathbf{y} + \mathbf{z}$ of a true underlying image $\mathbf{y}$. The Bayesian might assume that $\mathbf{y}$ has PDF $p(\mathbf{y})$ and that the noise $\mathbf{z}$ is independent of $\mathbf{y}$, with probability model $q(\mathbf{z})$. Then the most likely reconstruction (maximum a posteriori) would solve

$$\max_{\mathbf{y}} p(\mathbf{y}) q(\tilde{\mathbf{y}} - \mathbf{y}). \tag{46}$$

This approach has been tried successfully in many concrete problems, with a wide range of interesting results.[2]

Finding prior distributions for signals has been a very active topic of research in the signal and image processing communities. A familiar starting point takes a

[2] We have no quarrel with orthodox Bayesian approaches. However, we do insist on making a distinction between the practical approach of using Bayes to quickly derive candidate algorithms, and the orthodox approach of believing that the assumptions are strictly true and the correctness of the assumptions is the unique reason that such algorithms can work. The results discussed earlier in this paper contradict orthodoxy: they provide a different explanation as to why some important Bayesian algorithms are successful in certain situations where sparsity is present. Bayesians can be right for the wrong reasons!

Gaussian prior on the signal, for example, $p(\mathbf{y}) \propto \exp(-\lambda\|\mathbf{L}\mathbf{y}\|_2^2)$, where $\mathbf{L}$ is the discrete Laplacian. Such Gaussian priors are frequently used and, of course, are intimately connected to beautiful classical topics such as Wiener filtering [95].

Gaussian processes have many beautiful analytic properties; however, they fail to match empirical facts. Edges are fundamental components of image content, and yet stationary Gaussian processes fail to exhibit them properly. It has been repeatedly found that using non-Gaussian priors often gives much better results in the Bayesian framework; an example is obtained by replacing the $\ell_2$ norm by the $\ell_1$ norm in the exponent of the prior density $p(\mathbf{y})$, and the second-order Laplacian operator by a pair of first-order difference operators (horizontal $\mathbf{D}_h$ and vertical $\mathbf{D}_v$), one in each direction:

$$p(\mathbf{y}) \propto \exp\left(-\lambda(\|\mathbf{D_h}\mathbf{y}\|_1 + \|\mathbf{D_v}\mathbf{y}\|_1)\right). \tag{47}$$

Another approach uses signal transforms. Let $\mathbf{\Psi}$ be the matrix associated with a discrete orthogonal wavelet transform, i.e., a matrix whose columns are the orthogonal basis functions in a specific wavelet transform (say, Daubechies nearly symmetric, with 8 vanishing moments) [116]. Consider a prior

$$p(\mathbf{y}) \propto \exp(-\lambda\|\mathbf{\Psi}^T\mathbf{y}\|_1). \tag{48}$$

Bayesian methods with such priors often outperform traditional Gaussian priors in image denoising [96].

The specific non-Gaussian priors we just mentioned give rise to interesting signal processing algorithms under the MAP framework. Writing the problem in terms of MAP estimation (as in (46)) gives for the prior in (47),

$$\min_{\mathbf{y}} \;\; \frac{1}{2}\|\tilde{\mathbf{y}} - \mathbf{y}\|_2^2 + \lambda\|\mathbf{D_h}\mathbf{y}\|_1 + \lambda\|\mathbf{D_v}\mathbf{y}\|_1, \tag{49}$$

while for the prior in (48),

$$\min_{\mathbf{y}} \;\; \frac{1}{2}\|\tilde{\mathbf{y}} - \mathbf{y}\|_2^2 + \lambda\|\mathbf{\Psi}^T\mathbf{y}\|_1. \tag{50}$$

We now step out of the Bayesian framework and interpret these optimization criteria as algorithm generators. We recognize that the first generates a variant of total-variation denoising [141, 41], while the second generates an instance of wavelet denoising—both very successful algorithms with hundreds of application papers in print.

The frequent success of these algorithms causes difficulties for Bayesian interpretation. The corresponding prior distributions are demonstrably not in agreement with image statistics in those cases where these algorithms are most successful. What *is* true is that in those cases where such algorithms are dramatically most successful, there is an underlying *transform sparsity* of the image content: in the case of total variation this means that the spatial gradients are nearly zero in most pixels, while in the case of wavelet denoising this means that most wavelet coefficients are nearly zero. In each successful case, the algorithm involves a transform which takes the signal and renders it sparse.

An orthodox Bayesian would in such cases seek a better prior. Careful empirical modeling of wavelet coefficients of images with edges has shown that, in many cases, the prior model $p(\mathbf{y}) \propto \exp(-\lambda\|\mathbf{T}\mathbf{y}\|_1)$ can indeed be improved [35, 144, 10]. The general form $p(\mathbf{y}) \propto \exp(-\lambda \|\mathbf{T}\mathbf{y}\|_r^r)$ with $0 < r < 1$ has been studied, and values of $r$ significantly smaller than 1 have been found to give a better fit to image libraries than $r = 1$. Surprisingly, however, the actual algorithm that results from this "better" model is not qualitatively different than $\ell_1$ minimization. Furthermore, the performance of the two algorithms, when the underlying noiseless object truly has sparse wavelet coefficients, is comparable.

Such observations suggest that a driving factor in content modeling is *sparsity.* If a given transform $\mathbf{T}$ maps the content into a sparse vector, that matters a great deal. The precise amplitude distribution of the nonzeros in such a transform domain may be a detail which matters very little in comparison.

**4.2. Combined Representation.** Continuing this line of thinking, we ask, if sparsity is so important and fundamental, what is the best way to achieve it? Traditional transform techniques do achieve some success in sparsifying image content, but they may not be the best one can do. Traditional transform techniques apply a linear transform $\mathbf{T}$ to the signal content and place a prior on the resulting transformed content.

Signal content is often a mixture of several different types of phenomena: harmonics and transients in acoustic data, and edges and textures in image data. Each component of such a mixture may be modeled in its own adapted fashion. A harmonic subsignal can be modeled by a superposition of sinusoids, while a transient might be modeled by a superposition of spikes.

Considerations of sparsity play out differently in this setting. If we restrict ourselves to using a single representation, say, sinusoids, we do not expect such a mixture of sinusoids and transients to be sparsely representable from sinusoids alone. In fact, the uncertainty principle given earlier expressly prevents this! To achieve sparsity we must combine several representations. Suppose we have two bases with corresponding matrices $\mathbf{\Psi}$ and $\mathbf{\Phi}$ that have as columns the elements of each basis. Then we model the signal of interest as a superposition of elements from each basis:

$$\mathbf{y} = \mathbf{\Psi}\mathbf{u} + \mathbf{\Phi}\mathbf{v}. \tag{51}$$

Here the vectors $\mathbf{u}$ and $\mathbf{v}$ give coefficients allowing us to represent $\mathbf{y}$.

This leads us to an important distinction. In harmonic analysis, the operation of transforming from a signal domain into a transform domain is called *analysis.* The operation of returning from a transform domain to a signal domain is called *synthesis.* We are trying to synthesize a signal from a combined transform domain; the synthesis coefficients provide a combined representation of the signal. In this setting, sparsity means that the combined coefficient vector $(\mathbf{u}, \mathbf{v})$ is sparse.

This viewpoint is forced on us by the uncertainty principle, which says *both* that sparse *analysis* coefficients cannot be expected in general and that sufficiently sparse *synthesis* coefficients can be uniquely recovered in some settings.

**4.3. A Generative Model.** Here is a simple way to sparsely synthesize signals. Start from an $n \times m$ matrix $\mathbf{A}$ whose columns are the elementary "atoms" of our model, and generate at random a vector $\mathbf{x}$ having $m$ entries, only $k_0$ of which are nonzero. Choose the positions of the nonzeros uniformly at random, with the values of the nonzeros chosen from a Laplace distribution having PDF $p(x|x \neq 0) = \exp(-\alpha|x|)/(2\alpha)$. For example, let $\mathbf{A} = [\mathbf{I}\ \mathbf{F}]$ be the combined time-frequency dictionary, so that $m = 2n$, and let $k_0 = n/10$, so that only a small fraction of the entries in $\mathbf{x}$ are nonzero. In this way, we generate a random sparse combination of spikes and sinusoids.

As another example, let $\mathbf{A}$ be the concatenation $[\mathbf{F},\ \mathbf{W}]$ of the Fourier matrix and the matrix of an orthonormal wavelet transform (say, the Daubechies $D_4$ wavelets). We get a random combination of harmonic signals and transients.

This overall approach provides us with a flexible class of probabilistic models. By varying the dictionary $\mathbf{A}$, the sparsity control parameter $k_0$, and the amplitude parameter $\alpha$, we get different signal types with markedly different characteristics.

Of course, "real signals" are expected to deviate from this model. For instance, they are expected to contain at least some noise. Our model can incorporate this effect by adding a noise vector $\mathbf{z}$, uniformly distributed on the sphere of radius $\epsilon$. The final generated noisy signal is $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{z}$. Call this class of models $\mathcal{M}(\mathbf{A}, k_0, \alpha, \epsilon)$; informally, these are Sparse-Land signals. Many natural variations are possible:

- One might instead have a random number of nonzeros, and so, for example, sample $k_0$ from a geometric or a Poisson distribution. In cases where $n$ is large, this variation would make little difference.
- One might instead use Gaussian white noise of power $\sigma^2$. In cases where $n$ is large, this is almost the same thing, provided we calibrate parameters using $\epsilon^2 = n\sigma^2$.
- One might be concerned about the specific choice of the Laplace density for the nonzeros. As we argued earlier, for the specific phenomena we are concerned with, getting this distribution right doesn't much matter, as surprising as that may seem.
- One might allow the sparsity parameter to be position dependent, for example, to control the sparsity in blocks. Thus, with the wavelet transform, it makes sense to allow coarse scales to be rather dense, while fine scales get increasingly sparse as the spatial scale shrinks.

While these possibilities are important for empirical work, in this exposition we work with the simpler model not having these features.

## 5. Processing of Sparsely Generated Signals.

How do we practice signal processing in Sparse-Land? Suppose we have a signal $\mathbf{y}$ which has been generated from the model $\mathcal{M}(\mathbf{A}, k_0, \alpha, \epsilon)$ and the parameters of the model are known. There are numerous signal processing tasks that could be of interest to us; let's discuss them and see how sparsity-seeking representations might enter in.

### 5.1. Possible Core Applications.

- **Analysis.** Given $\mathbf{y}$, can we determine the underlying vector $\mathbf{x}_0$ which generated it? This process may be called *atomic decomposition*, as it leads us to infer the individual atoms that actually generated $\mathbf{y}$. Clearly, the true underlying representation obeys $\|\mathbf{A}\mathbf{x}_0 - \mathbf{y}\|_2 \le \epsilon$, but there will be many other vectors $\mathbf{x}$ generating similarly good approximations to $\mathbf{y}$. Suppose we can actually solve $(P_0^\epsilon)$:

$$(P_0^\epsilon): \qquad \min_{\mathbf{x}} \ \|\mathbf{x}\|_0 \ \text{ subject to } \ \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \le \epsilon.$$

  Under our assumptions, the solution $\mathbf{x}_0^\epsilon$ of this problem, though not necessarily the underlying $\mathbf{x}_0$, will also be sparse and have $k_0$ or fewer nonzeros. Our earlier results show that if $k_0$ is small enough, then the solution of $(P_0^\epsilon)$ is at most $O(\epsilon)$ away from $\mathbf{x}_0$.
- **Compression.** Nominally $\mathbf{y}$ requires a description by $n$ numbers. However, if we can solve $(P_0^\delta)$ where $\delta \ge \epsilon$, then the resulting solution $\mathbf{x}_0^\delta$ affords an approximation $\hat{\mathbf{y}} = \mathbf{A}\mathbf{x}_0^\delta$ to $\mathbf{y}$ using at most $k_0$ scalars, with an approximation

error at most $\delta$. By increasing $\delta$ we obtain stronger compression with larger approximation error, and in this way we obtain a rate-distortion curve for a compression mechanism.

- **Denoising.** Suppose that we observe not $\mathbf{y}$, but instead a noisy version $\tilde{\mathbf{y}} = \mathbf{y} + \mathbf{v}$, where the noise is known to obey $\|\mathbf{v}\|_2 \leq \delta$. If we can solve $(P_0^{\delta+\epsilon})$, then the resulting solution $\mathbf{x}_0^{\delta+\epsilon}$ will have at most $k_0$ nonzeros; our earlier results show that if $k_0$ is small enough, then $\mathbf{x}_0^{\epsilon+\delta}$ is at most $O(\epsilon + \delta)$ away from $\mathbf{x}_0$.
- **Inverse Problems.** Even more generally, suppose that we observe not $\mathbf{y}$, but a noisy indirect measurement of it, $\tilde{\mathbf{y}} = \mathbf{H}\mathbf{y} + \mathbf{v}$. Here the linear operator $\mathbf{H}$ generates blurring, masking, or some other kind of degradation, and $\mathbf{v}$ is noise as before. If we could solve

  $$\min_{\mathbf{x}} \ \|\mathbf{x}\|_0 \ \text{ subject to } \ \|\tilde{\mathbf{y}} - \mathbf{H}\mathbf{A}\mathbf{x}\|_2 \leq \delta + \epsilon,$$

  then we would expect to identify directly the sparse components of the underlying signal and obtain an approximation $\mathbf{A}\mathbf{x}_0^{\delta+\epsilon}$.
- **Compressed Sensing.** For signals which are sparsely generated, one can obtain good reconstructions from reduced numbers of measurements—thereby compressing the *sensing process* rather than the traditionally sensed *data*. In fact, let $\mathbf{P}$ be a random $j_0 \times n$ matrix with Gaussian i.i.d. entries, and suppose that it is possible to directly measure $\mathbf{c} = \mathbf{P}\mathbf{y}$, which has $j_0$ entries, rather than $\mathbf{y}$, which has $n$. Attempt recovery by solving

  $$\min_{\mathbf{x}} \ \|\mathbf{x}\|_0 \ \text{ subject to } \ \|\mathbf{c} - \mathbf{P}\mathbf{A}\mathbf{x}\|_2 \leq \epsilon$$

  to obtain the sparse representation and then synthesizing an approximate reconstruction using $\mathbf{A}\mathbf{x}_0^{\epsilon}$ [18, 15, 17, 58, 42].
- **Morphological Component Analysis (MCA).** Suppose that the observed signal is a superposition of two different subsignals $\mathbf{y}_1$, $\mathbf{y}_2$ (i.e., $\mathbf{y} = \mathbf{y}_1 + \mathbf{y}_2$), where $\mathbf{y}_1$ is sparsely generated using model $\mathcal{M}_1$ and $\mathbf{y}_2$ is sparsely generated using model $\mathcal{M}_2$. Can we separate the two sources? Such source separation problems are fundamental in the processing of acoustic signals, for example, in the separation of speech from impulsive noise by *independent component analysis* (ICA) algorithms [94, 164, 109]. Turning to the signal model presented here, if we could solve

  $$\min_{\mathbf{x}_1, \mathbf{x}_2} \ \|\mathbf{x}_1\|_0 + \|\mathbf{x}_2\|_0 \ \text{ subject to } \ \|\mathbf{y} - \mathbf{A}_1\mathbf{x}_1 - \mathbf{A}_2\mathbf{x}_2\|_2^2 \leq \epsilon_1^2 + \epsilon_2^2,$$

  the resulting solution $(\mathbf{x}_1^{\epsilon}, \mathbf{x}_2^{\epsilon})$ would generate a plausible solution $\hat{\mathbf{y}}_1 = \mathbf{A}\mathbf{x}_1^{\epsilon}$, $\hat{\mathbf{y}}_2 = \mathbf{A}\mathbf{x}_2^{\epsilon}$ to the separation problem. In fact, there have been several successful trials of this idea, first in acoustic and later in image processing [122, 148, 147, 7, 8]. An appealing image processing application that relies on MCA is inpainting, where missing pixels in an image are filled in, based on a sparse representation of the existing pixels [68]. MCA is necessary because the piecewise smooth (cartoon) and texture contents of the image must be separated as part of this recovery process. See [68] for more details.

A wide range of other applications including encryption, watermarking, scrambling, target detection, and more, can also be envisioned. All these applications call for the solution of $(P_0^{\delta})$, or variants. We have intentionally described all these proposed applications in the conditional mood, since in general $(P_0^{\delta})$ is not known to be tractable or even well defined. However, our earlier discussion of $(P_0^\delta)$ shows that the problem, under the right conditions, is sensible and can be approximately solved by practical algorithms.

A word of caution is required: In any serious application we should check whether the dictionary $\mathbf{A}$ and the sparsity level $k_0$ are suitable for application of existing results, or whether new results are perhaps needed, and, similarly, whether the algorithms we have discussed work well, or whether new algorithms need to be designed. Again, in general, $(P_0^\delta)$ is not a well-defined problem, so suitability for a given application must always be verified.

Since at this stage the reader may be skeptical that applications inspired by solving $(P_0)$ and $(P_0^\delta)$ can really work, we present in Figures 5 and 6 two worked out large-scale applications.

Figure 5 presents compressed sensing of dynamic MRI—real-time acquisition of heart motion—by Michael Lustig and coworkers at the Stanford MRI lab [112, 111]. They obtain a successful reconstruction of moving imagery of the beating heart from raw pseudorandom samples of the k-t space, with a factor of 7 undersampling, i.e., they solve a system of equations which has seven times more unknowns than equations. Sparsity of the desired solution in the wavelet-Fourier domain is exploited by $\ell_1$ minimization.

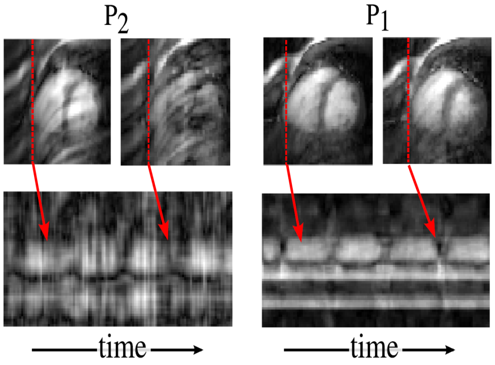
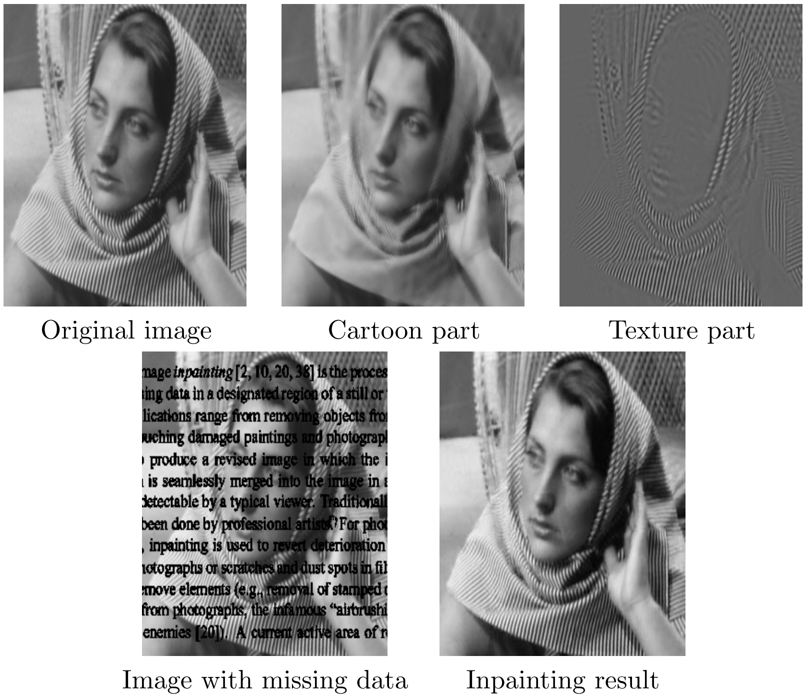


**Fig. 5** *Compressed sensing in dynamic acquisition of heart motion. A dynamic image is created in a setting where the classical sampling theorem allows reconstruction at rates no higher than* 3.6 *frames per second. Attempts to reconstruct at faster frame rates using classical linear tools (based on $\ell_2$ minimization) fail badly, exhibiting temporal blurring and artifacts (see panel $P_2$). By instead using $\ell_1$ penalized reconstruction on the sparse transform coefficients, the dynamic sequence can be reconstructed at the much higher rate of* 25 *frames per second with significantly reduced image artifacts (see panel $P_1$). The top images show the heart at two time frames, and the bottom ones present the time series of a cross section of the heart. More information can be found in* [112, 111].

Figure 6 presents an image separation result obtained by Jean-Luc Starck and coworkers [148, 147], where the image `Barbara` is decomposed into piecewise smooth (cartoon) and texture, using MCA as described above. They used a dictionary combining two representations: curvelets [146, 12, 13] for representing the cartoon part, and local overlapped DCT for the texture. The second row in this figure, taken from [68], presents inpainting results, where missing values (the text) are recovered, based on the above separation. We see again a successful application driven by the goal of approximately solving $(P_0^\delta)$.


Original image

Cartoon part

Texture part

Image with missing data

Inpainting result

nage *inpainting* [2, 10, 20, 38] is the proces
ing data in a designated region of a still or
lications range from removing objects fro
uching damaged paintings and photograp
produce a revised image in which the i
is seamlessly merged into the image in
detectable by a typical viewer. Traditionall
been done by professional artists. For pho
, inpainting is used to revert deterioration
otographs or scratches and dust spots in fil
emove elements (e.g., removal of stamped
from photographs, the infamous "airbrushi
enemies [20]). A current active area of r

**Fig. 6** *Top row: MCA for image separation to texture and cartoon* [148, 147]. *Bottom row: Image inpainting—filling in missing pixels (the text) in the image* [68].

### 5.2. The Quest for a Dictionary.

A fundamental ingredient in the definition of Sparse-Land's signals and the deployment to applications is the dictionary $\mathbf{A}$. How can we wisely choose $\mathbf{A}$ to perform well on the signals we have in mind? One line of work considered choosing preconstructed dictionaries, such as undecimated wavelets [149], steerable wavelets [145, 37, 136], contourlets [38, 39, 40, 70, 71], curvelets [146, 12], and others [22, 123]. These are generally suitable for stylized "cartoon-like" image content, assumed to be piecewise smooth and with smooth boundaries. Some of these papers provide detailed theoretical analysis establishing the sparsity of the representation coefficients for such content.

Alternatively, one can use a tunable selection, in which a basis or frame is generated under the control of particular parameter (discrete or continuous): wavelet packets (parameter is time-frequency subdivision) [28, 29, 121] or bandelettes (parameter is spatial partition) [105, 117]. A third option is to build a *training database* of signal instances similar to those anticipated in the application, and build an empirically-learned dictionary, in which the generating atoms come from the underlying empirical data rather than from some theoretical model; such a dictionary can then be used in the application as a fixed and redundant dictionary. We now explore this third option in detail.

Assume that a training database $\{\mathbf{y}_i\}_{i=1}^{M}$ is given, thought to have been generated by some fixed but unknown model $\mathcal{M}_{\{\mathbf{A},k_0,\alpha,\epsilon\}}$. Can this training database allow us to identify the generating model, specifically the dictionary $\mathbf{A}$? This rather difficult problem was studied initially by Olshausen and Field [128, 126, 127], who were motivated by an analogy between the atoms of a dictionary and the population of simple cells in the visual cortex. They thought that learning a dictionary empirically might model evolutionary processes that led to the existing collection of simple cells, and they indeed were able to find a rough empirical match between the properties of a learned dictionary and some known properties of the population of simple cells.

Later work extended their methodology and algorithm in various forms [107, 108, 69, 101, 106, 3, 2]. Here we describe a related training mechanism based on [69, 101, 3, 2].

Assume that $\epsilon$—the model deviation—is known, and that our aim is the estimation of $\mathbf{A}$. Consider the following optimization problem:

$$(52) \qquad \min_{\mathbf{A},\{\mathbf{x}_i\}_{i=1}^M} \sum_{i=1}^{M} \|\mathbf{x}_i\|_0 \quad \text{subject to} \quad \|\mathbf{y}_i - \mathbf{A}\mathbf{x}_i\|_2 \le \epsilon, \quad 1 \le i \le M.$$

If we could solve this, then we would obtain the dictionary $\mathbf{A}$ that gives us a sparse approximation to all $M$ elements of our training set. The roles of the penalty and the constraints in (52) might also be reversed if we choose to constrain the sparsity and obtain the best fit for that sparsity, and if we assume that $k_0$ is known:

$$(53) \qquad \min_{\mathbf{A},\{\mathbf{x}_i\}_{i=1}^M} \sum_{i=1}^{M} \|\mathbf{y}_i - \mathbf{A}\mathbf{x}_i\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}_i\|_0 \le k_0, \ 1 \le i \le M.$$

Are these two problems properly posed? Do they have meaningful solutions? There are some obvious indeterminacies (scaling and permutation of the columns). If we fix a scale and ordering, it is still unclear whether there is a meaningful answer in general.

Is there a uniqueness property underlying this problem, implying that only one dictionary exists, such that it explains sparsely the set of training vectors? Surprisingly, at least for the case $\epsilon = 0$, there can be an answer to this problem, as shown in [2]. Suppose there exists a dictionary $\mathbf{A}_0$ with $spark(\mathbf{A}_0) > 2$ and a sufficiently diverse database of examples, all of which are representable using at most $k_0 < spark(\mathbf{A}_0)/2$ atoms. Then any other dictionary $\mathbf{A}$ that permits an equally sparse representation of all the elements of the training database is derivable from $\mathbf{A}_0$ simply by rescaling or permutation of the columns of $\mathbf{A}_0$.

Some readers may prefer to think in terms of matrix factorizations. Concatenate all the database vectors columnwise, forming an $n \times M$ matrix $\mathbf{Y}$, and, similarly, all the corresponding sparse representations into a matrix $\mathbf{X}$ of size $m \times M$; thus the dictionary satisfies $\mathbf{Y} = \mathbf{A}\mathbf{X}$. The problem of discovering an underlying dictionary is thus the same as the problem of discovering a factorization of the matrix $\mathbf{Y}$ as $\mathbf{A}\mathbf{X}$, where $\mathbf{A}$ and $\mathbf{X}$ have the indicated shapes and $\mathbf{X}$ has sparse columns. The matrix factorization viewpoint connects this problem with related problems of nonnegative matrix factorization [104, 55] and sparse nonnegative matrix factorization [92, 1].

Clearly, there is no general practical algorithm for solving problem (52) or (53), for the same reasons that there is no general practical algorithm for solving $(P_0)$, only more so! However, just as with $(P_0)$, the lack of general guarantees is no reason not to try heuristic methods and see how they do in specific cases.

We can view the problem posed in (53) as a nested minimization problem: an inner minimization of the number of nonzeros in the representation vectors $\mathbf{x}_i$, for a given fixed $\mathbf{A}$ and an outer minimization over $\mathbf{A}$. A strategy of alternating minimization thus seems to us very natural; at the $k$th step, we use the dictionary $\mathbf{A}_{(k-1)}$ from the $(k-1)$th step and solve $M$ instances of $(P_0^\epsilon)$, one for each database entry $\mathbf{y}_i$, each using the dictionary $\mathbf{A}_{(k-1)}$. This gives us the matrix $\mathbf{X}_{(k)}$, and we then solve for $\mathbf{A}_{(k)}$ by least squares, so that

$$\mathbf{A}_{(k)} = \arg\min_{\mathbf{A}} \ \|\mathbf{Y} - \mathbf{A}\mathbf{X}_{(k)}\|_F^2 = \mathbf{Y}\mathbf{X}_{(k)}^T \left(\mathbf{X}_{(k)}\mathbf{X}_{(k)}^T\right)^{-1}. \tag{54}$$

We may also rescale the columns of the dictionary obtained. We increment $k$ and unless we have satisfied a convergence criterion, we repeat the above loop. Such a block-coordinate descent algorithm was proposed in [69, 101] and termed the method of directions (MOD).

An improved update rule for the dictionary can be proposed, where the atoms (i.e., columns) in $\mathbf{A}$ are handled sequentially. This leads to the K-SVD algorithm, as developed and demonstrated in [3, 2]. Keeping all the columns fixed apart from the $j_0$th one, $\mathbf{a}_{j_0}$, this column can be updated along with the coefficients that multiply it in $\mathbf{X}$. The term to be minimized—see (54)—can be rewritten as[3]

$$\|\mathbf{Y} - \mathbf{A}\mathbf{X}\|_2^2 = \left\|\mathbf{Y} - \sum_{j=1}^{m} \mathbf{a}_j \mathbf{x}_j^T\right\|_2^2 = \left\|\left(\mathbf{Y} - \sum_{j\neq j_0} \mathbf{a}_j \mathbf{x}_j^T\right) - \mathbf{a}_{j_0}\mathbf{x}_{j_0}^T\right\|_2^2. \tag{55}$$

In this description, $\mathbf{x}_j^T$ stands for the $j$th row from $\mathbf{X}$. In the above expression we target the update of both $\mathbf{a}_{j_0}$ and $\mathbf{x}_{j_0}^T$, referring to the term

$$\mathbf{E}_{j_0} = \mathbf{Y} - \sum_{j\neq j_0} \mathbf{a}_j \mathbf{x}_j^T \tag{56}$$

as a known precomputed error matrix.

The optimal $\mathbf{a}_{j_0}$ and $\mathbf{x}_{j_0}^T$ minimizing (55) are furnished by an SVD (rank-1 approximation [83]), but this typically yields a dense vector $\mathbf{x}_{j_0}^T$. In order to minimize this term while fixing the cardinalities of all representations, a subset of the columns of $\mathbf{E}_{j_0}$ should be taken—those that correspond to the signals from the example set that are using the $j_0$th atom, namely, those columns where $\mathbf{x}_{j_0}^T$ are nonzero. For this submatrix a rank-1 approximation via SVD [83] can be applied, updating both the atom $\mathbf{a}_{j_0}$ and the coefficients that deploy it in the sparse representations. This dual update leads to a substantial speedup in the convergence of the training algorithm.

Interestingly, if the above process is considered for the case where $k_0 = 1$, constraining the representation coefficients to be binary (1 or 0), the above-posed problem reduces to a clustering task. Furthermore, in such a case the above training algorithms simplify to the well-known K-means algorithm [81]. While each iteration of K-means computes means over $K$ different subsets, the K-SVD algorithm performs the SVD over each of $K$ different submatrices, hence the name K-SVD ($K$ is assumed to be the number of columns in $\mathbf{A}$ in [3, 2]). Exhibit 4 describes the MOD and the K-SVD algorithms in detail.

## 6. Applications in Image Processing.

The sparse representation viewpoint discussed so far is merely that—a viewpoint. The theoretical results we have given merely tell us that sparse modeling is, in favorable cases, a mathematically well-founded enterprise with practically useful computational tools. The only way to tell whether sparse modeling works in the real world is to apply it and see how it performs!

[3] To simplify notation, we now omit the iteration number $k$.

```
Task: Train a dictionary $\mathbf{A}$ to sparsely represent the data $\{\mathbf{y}_i\}_{i=1}^M$ by approximating the solution
      to the problem posed in (53).
Initialization: Initialize $k = 0$, and
    • Initialize Dictionary: Build $\mathbf{A}_{(0)} \in \mathbb{R}^{n\times m}$, either by using random entries or
      using $m$ randomly chosen examples.
    • Normalization: Normalize the columns of $\mathbf{A}_{(0)}$.
Main Iteration: Increment $k$ by $1$, and apply
    • Sparse Coding Stage: Use a pursuit algorithm to approximate the solution of
          $\hat{\mathbf{x}}_i = \arg\min_{\mathbf{x}} \ \|\mathbf{y}_i - \mathbf{A}_{(k-1)}\mathbf{x}\|_2^2$ subject to $\|\mathbf{x}\|_0 \le k_0,$
      obtaining sparse representations $\hat{\mathbf{x}}_i$ for $1 \le i \le M$. These form the matrix $\mathbf{X}_{(k)}$.
    • Dictionary Update Stage: Use one of the following options:
        – MOD: Update the dictionary by the formula
              $\mathbf{A}_{(k)} = \arg\min_{\mathbf{A}} \ \|\mathbf{Y} - \mathbf{A}\mathbf{X}_{(k)}\|_F^2 = \mathbf{Y}\mathbf{X}_{(k)}^T \left(\mathbf{X}_{(k)}\mathbf{X}_{(k)}^T\right)^{-1}.$
        – K-SVD: Use the following procedure to update the columns of the dictionary
          and obtain $\mathbf{A}_{(k)}$: Repeat for $j_0 = 1, 2, \ldots, m,$
            * Define the group of examples that use the atom $\mathbf{a}_{j_0}$,
                  $\Omega_{j_0} = \{i |\ 1 \le i \le M,\ \mathbf{X}_{(k)}[j_0, i] \ne 0\}.$
            * Compute the residual matrix
                  $\mathbf{E}_{j_0} = \mathbf{Y} - \sum_{j \ne j_0} \mathbf{a}_j \mathbf{x}_j^T,$
              where $\mathbf{x}^j$ are the $j$th rows in the matrix $\mathbf{X}_{(k)}$.
            * Restrict $\mathbf{E}_{j_0}$ by choosing only the columns corresponding to $\Omega_{j_0}$, and
              obtain $\mathbf{E}_{j_0}^R$.
            * Apply SVD decomposition $\mathbf{E}_{j_0}^R = \mathbf{U}\mathbf{\Delta}\mathbf{V}^T$. Update the dictionary atom
              $\mathbf{a}_{j_0} = \mathbf{u}_1$ and the representations by $\mathbf{x}_R^{j_0} = \mathbf{\Delta}[1,1] \cdot \mathbf{v}_1$.
    • Stopping Rule: If $\|\mathbf{Y} - \mathbf{A}_{(k)}\mathbf{X}_{(k)}\|_F^2$ is smaller than a preselected threshold, stop.
      Otherwise, apply another iteration.
Output: The desired result is $\mathbf{A}_{(k)}$.
```

**Exhibit 4.** *The MOD* [69, 101] *and K-SVD* [3, 2] *dictionary-learning algorithms.*

In this section, we review selected results applying this viewpoint in image compression and image denoising. Due to space imitations, we are unable to discuss many other interesting examples, including problems in array processing [114, 115], inpainting in images [68, 88, 89, 72], image decomposition to cartoon and texture [122, 148, 147], and others [99, 7, 113, 137]. Also, the applications presented here rely on an adapted dictionary using the K-SVD algorithm, but successful applications can be demonstrated using other dictionaries, such as curvelets, contourlets, and others; see [146, 148, 68, 66, 67] for some examples.

## 6.1. Compression of Facial Images.

### 6.1.1. The Application.

Image compression is fundamental to today's use of digital imagery; we exploit it on a daily basis, in our digital cameras, satellite TVs and Internet downloads. Sparse representation already lies behind many successful applications of image compression; the JPEG and JPEG-2000 compression standards exploit the fact that natural images have sparse representations in the Fourier and wavelet domains, respectively. (Of course, sparsity alone is not enough to develop an effective content transmission system; in particular, efficient coding of sparse vectors is needed in order to obtain bit streams.)

Because images are becoming so commonplace, we now often hear of highly targeted applications for imaging—biometric identification, fingerprint searching, cardiac imaging, and so on. Each such specialized application raises the issue of *application-specific* compression. Rather than use a generic representation, such as the Fourier or wavelet transform, one employs a dictionary which is specific to the image content encountered in a given application.

In this section we address the compression of facial images, considering the application of passport photograph storage in a digital ID system.

**6.1.2. Methodology and Algorithms.** We gather a passport photo database of 2600 facial images of size $180 \times 220$ pixels to train and test the compression algorithms to be described shortly. We consider two types of compression algorithms: fixed transform-based algorithms based on DCT (JPEG) and DWT (JPEG-2000),[4] and two content-adaptive algorithms based on learned dictionaries—principal component analysis (PCA) and K-SVD.

Both adapted methods use dictionaries that are learned based on disjoint image patches of size $15 \times 15$ pixels that are extracted from 2500 training images in the database. Thus, every location in the image obtains a different dictionary based on its content, as manifested in these 2500 examples.

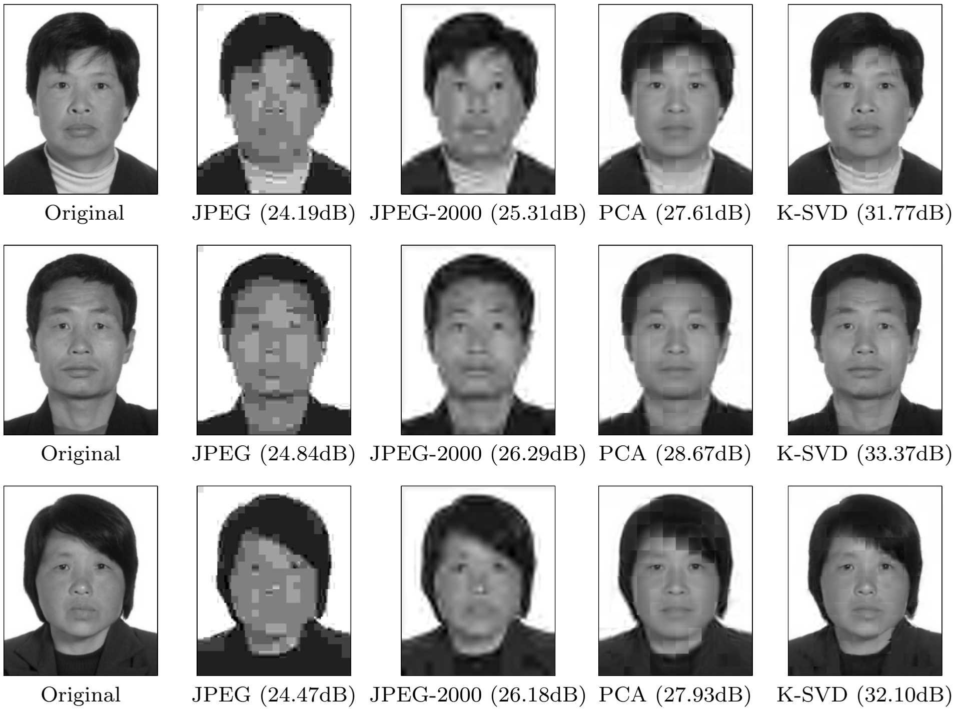
The PCA technique models each patch as a realization of a multivariate Gaussian distribution, and the learned dictionary is simply the set of usual principal axes based on an empirical covariance matrix. The K-SVD technique instead models each patch as approximately a sparse linear combination of 512 atoms which have been learned from the image database. PCA is of course classical and relatively inexpensive to compute, while the K-SVD technique is more time consuming. K-SVD training using MATLAB requires $\approx 10$ hours. After the training, the compression/decompression of a facial image takes less than one second, using the stored dictionaries both at the encoder and at the decoder. More details on these and other experiments can be found in [9], and here we concentrate on showing the main results.

**6.1.3. Experiments and Results.** Each method was evaluated at two different compression ratios, corresponding to 550 and 880 bytes, respectively. The rendering of transform coefficients into byte streams was done very crudely, leaving open the possibility of further improvements. It is standard to evaluate performance using the peak signal-to-noise ratio (PSNR), defined by

$$PSNR = 20 \log \left\{ \frac{255}{\sqrt{\sum_{i,j \in \Omega} \left(y(i,j) - \hat{y}(i,j)\right)^2}} \right\},$$

where $y(i,j)$ are the original image pixels and $\hat{y}(i,j)$ are the compressed-decompressed ones.

[4] We used implementations of these methods available in IrfanView using default parameters. The file sizes include the headers, which in principle could be omitted to obtain better compression. More on this option and its impact can be found in [9].



Original JPEG (24.19dB) JPEG-2000 (25.31dB) PCA (27.61dB) K-SVD (31.77dB)

Original JPEG (24.84dB) JPEG-2000 (26.29dB) PCA (28.67dB) K-SVD (33.37dB)

Original JPEG (24.47dB) JPEG-2000 (26.18dB) PCA (27.93dB) K-SVD (32.10dB)

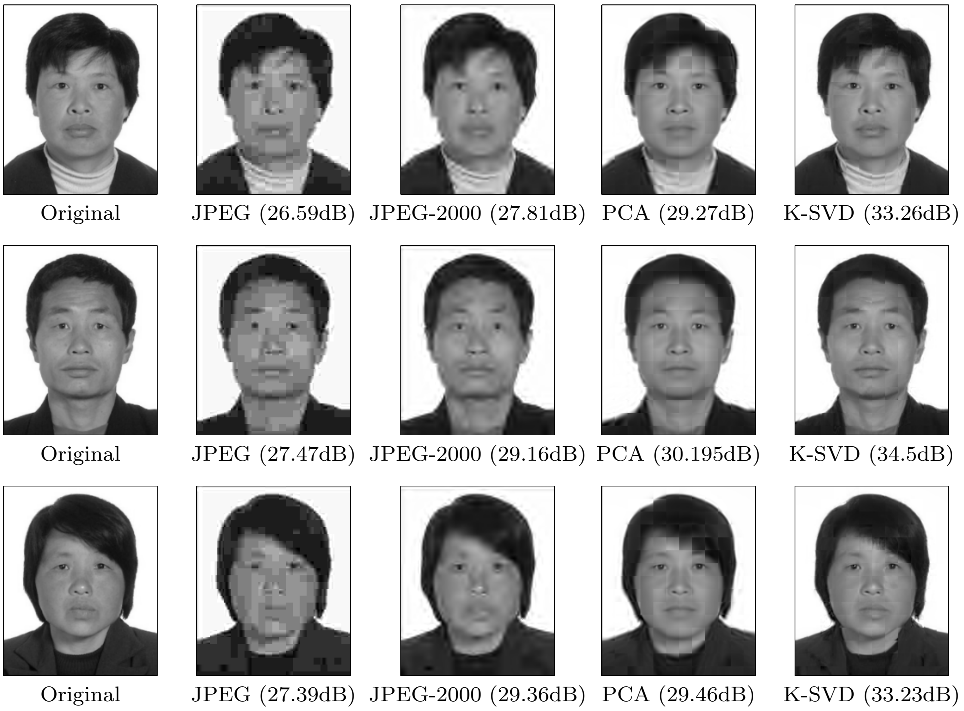
**Fig. 7** *Face image compression with* 550 *bytes per image: Comparison of results from JPEG, JPEG-*2000*, PCA, and sparse coding with K-SVD dictionary training. The values below each result show the PSNR.*

Figures 7 and 8 show results at 550 bytes and 820 bytes per image, respectively, testing three images from the test set (as opposed to the training set, used for learning the dictionaries). As can be seen, the K-SVD method is far better than the others, both in the image quality and in the PSNR. The block artifacts seen in the results are due to the block-based coding employed, and further improvement can be introduced by a selective smoothing of the block edges.

### 6.2. Denoising of Images.

**6.2.1. The Application.** Images often contain noise, which may arise due to sensor imperfection, poor illumination, or communication errors. Removing such noise is of great benefit in many applications, and a wide variety of techniques have been proposed, based on ideas as disparate as partial differential equations, local polynomial or spline fitting, filtering, hidden Markov models, and shrinkage of transform coefficients. An extensive comparison of the leading methods is given in [136].

Sparse representation can also be applicable for image denoising; in recent years, many researchers have developed and applied novel transforms which represent images more sparsely than traditional transforms from harmonic analysis, and one primary application area has been image denoising. The transforms—steerable wavelets, curvelets, and related direction-sensitive transforms—have the ability to more sparsely represent edges than do Fourier and wavelet methods. By shrinkage of transform coefficients followed by reconstruction, some reduction in image noise is observed, while edges are approximately preserved [103, 19, 20, 21, 146, 136, 70, 71, 88, 89].



Original JPEG (26.59dB) JPEG-2000 (27.81dB) PCA (29.27dB) K-SVD (33.26dB)

Original JPEG (27.47dB) JPEG-2000 (29.16dB) PCA (30.195dB) K-SVD (34.5dB)

Original JPEG (27.39dB) JPEG-2000 (29.36dB) PCA (29.46dB) K-SVD (33.23dB)

**Fig. 8** *Face image compression with* 820 *bytes per image: Comparison of results from JPEG, JPEG-*2000*, PCA, and sparse coding with K-SVD dictionary training. The values below each result show the PSNR.*

**6.2.2. Methodology and Algorithms.** The denoising methods described in [63, 64] take a different approach: by training a dictionary on the image content directly. One option is to use a standard library of clean images, e.g., the Corel library of 60,000 images, and develop a standard dictionary adapted to general images. A more ambitious goal is to develop a dictionary adapted to the problem at hand, learning the dictionary from the noisy image itself! Presumably this yields sparser representations and a more effective denoising strategy. In fact, papers [63, 64] apply the K-SVD algorithm as shown in Exhibit 4 to image patches carved out of the noisy image.

Because of the curse of dimensionality, learning structure from data rapidly becomes intractable as the dimension of the feature vector increases. Therefore, the K-SVD algorithm must be used with relatively small image patches—in the cited papers, $8 \times 8$ patches were used. The cited papers apply sparse representation to each such patch extracted from the image and, for each pixel, average the results from all patches containing that pixel.

**6.2.3. Experiments and Results.** The results reported in [63, 64] and reproduced below are the best we have seen. Figure 9 shows the two dictionaries obtained—the global one that is based on a group of 15 natural scene images, and the one adapted to the image `Barbara`. Both dictionaries have 256 atoms. The denoising results are demonstrated in Figure 10 for both methods. Results are reported again using the PSNR between the original image (prior to the additive noise) and the denoising result.

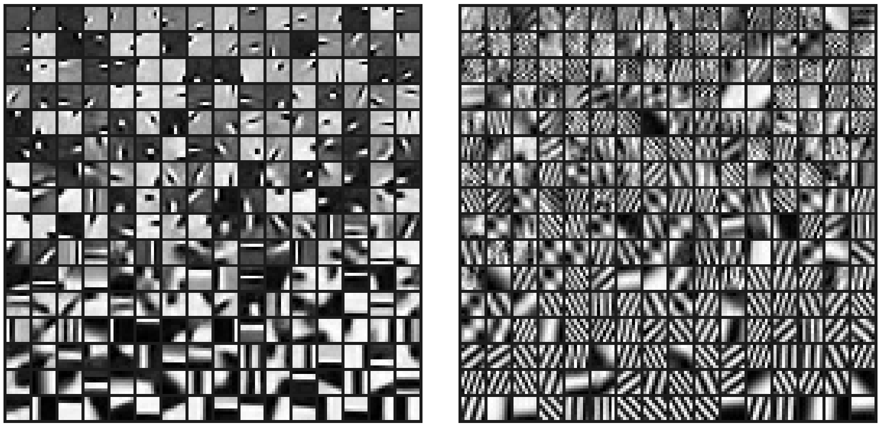
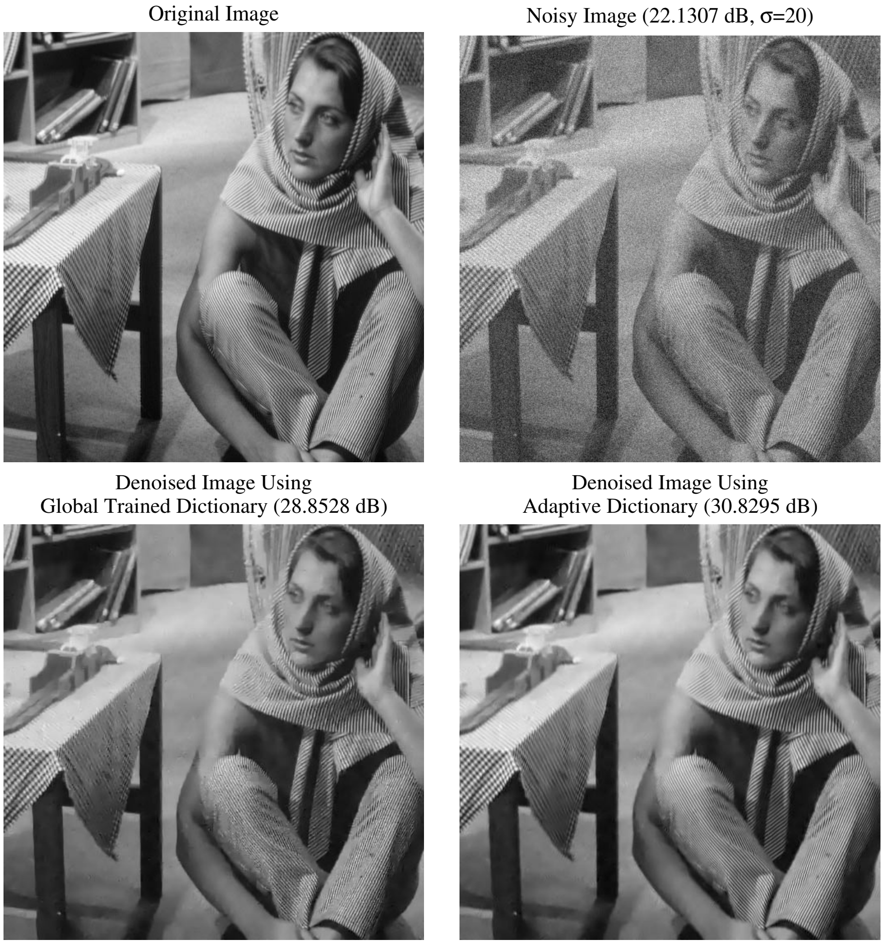


**Fig. 9** *Candidate dictionaries; The globally trained K-SVD dictionary for general images and the K-SVD dictionary trained on the noisy* `Barbara` *image directly.*



Original Image

Noisy Image (22.1307 dB, σ=20)

Denoised Image Using
Global Trained Dictionary (28.8528 dB)

Denoised Image Using
Adaptive Dictionary (30.8295 dB)

**Fig. 10** *Denoising comparisons: Additive noise standard deviation* $\sigma = 20$*; i.e., unprocessed* $PSNR = 22.13dB$*. The results using the globally trained and the adapted K-SVD dictionaries with patches of* $8 \times 8$ *show an improvement of* $6.7$ *and* $8.7dB$*, respectively.*

## 7. Concluding Remarks.

**7.1. Summary.** The fields of signal and image processing offer an unusually fertile playground for applied mathematicians, where the distance between an abstract mathematical idea and an application or even a product may be small. In this paper we have discussed the concept of sparse representations for signals and images. Although sparse representation is a poorly defined problem and a computationally impractical goal in general, we have pointed to mathematical results showing that under certain conditions one can obtain results of a positive nature, guaranteeing uniqueness, stability, or computational practicality. Inspired by such positive results we have explored the potential applications of sparse representation in real signal processing settings and shown that in certain denoising and compression tasks content-adapted sparse representation provides state-of-the-art solutions.

**7.2. Open Questions.** We list here several future research directions.

- Why do sparsity and redundancy as presented here form a good model for images? A theoretical/empirical claim that goes beyond the expected "try and see" would be very desirable.
- More work is required to carefully map the connections between the signal model studied in this paper and others (Markov random field (MRF), PCA, example-based regularization, and so on).
- A general purpose compression algorithm that leans on sparsity and redundancy? Watermarking? Encryption? Classification? All these and many more applications should be addressed to show the strength of the sparsity and redundancy concepts in representation.
- The presented model is not perfect, and this undermines its ability to further improve the performance of some applications. Model extensions to better match true data are desired. For example, defining the statistical dependencies within the representation coefficients is necessary.
- How can we synthesize signals based on the presented sparsity-redundancy model? The direct approach of randomly generating a sparse vector $\mathbf{x}$ with i.i.d. entries does not lead to natural images, even if the dictionary is of good quality. What modifications of this model are necessary to enable synthesis?
- Training the dictionary is limited to small signal dimensions. How can this limitation be circumvented? A multiscale concept seems to be natural in this context.
- Uniqueness or stability of the learned dictionaries has not been established. Empirically, the training also generates a denoising effect, but careful documentation of this effect and theoretical understanding are needed.
- How can the redundancy of the dictionary be chosen wisely? Is there a critical value above and below which performance deteriorates? Current applications tend to address this question empirically, and better understanding of the role of redundancy is required.
- In the dictionary training algorithms, can any algorithm be guaranteed to work under appropriate conditions, say, involving coherence? For example, if the desired dictionary is of known and small mutual coherence, and if the process is initialized by an arbitrary matrix with sufficiently small mutual coherence, is there a guaranteed path toward the desired dictionary?

**Acknowledgments.** Thanks to Yaacov Tsaig (Stanford) and Joel Tropp (CalTech) for extensive editorial suggestions—only a fraction of which could be accom-

modated. Thanks to Michael Lustig (Stanford) and Jean-Luc Starck (CEA-Saclay) for providing figures illustrating their work.

## REFERENCES

[1] M. Aharon, M. Elad, and A.M. Bruckstein, *K-SVD and its non-negative variant for dictionary design*, in Wavelet Applications in Signal and Image Processing XI, Proc. SPIE Conf. 5914, SPIE, Bellingham, WA, 2005, pp. 11.1–11.13.

[2] M. Aharon, M. Elad, and A.M. Bruckstein, *On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them*, J. Linear Algebra Appl., 416 (2006), pp. 48–67.

[3] M. Aharon, M. Elad, and A.M. Bruckstein, *K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representation*, IEEE Trans. Signal Process., 54 (2006), pp. 4311–4322.

[4] A. Barron, A. Cohen, W. Dahmen, and R.A. DeVore, *Approximation and learning by greedy algorithms*, Ann. Statist., 36 (2008), pp. 64–94.

[5] J. Bioucas-Dias, *Bayesian wavelet-based image deconvolution: A GEM algorithm exploiting a class of heavy-tailed priors*, IEEE Trans. Image Process., 15 (2006), pp. 937–951.

[6] A. Björner, M. Las Vergnas, B. Sturmfels, N. White, and G. M. Ziegler, *Oriented Matroids*, Encyclopedia Math. Appl. 46, Cambridge University Press, Cambridge, UK, 1993.

[7] J. Bobin, Y. Moudden, J.-L. Starck, and M. Elad, *Morphological diversity and source separation*, IEEE Signal Process. Lett., 13 (2006), pp. 409–412.

[8] J. Bobin, J.-L. Starck, M.J. Fadili, and Y. Moudden, *SZ and CMB reconstruction using generalized morphological component analysis*, Statist. Method., to appear.

[9] O. Bryt and M. Elad, *Compression of facial images using the K-SVD algorithm*, J.Vrs. Commun. Image Rep., 19 (2008), pp. 270–283.

[10] R.W. Buccigrossi and E.P. Simoncelli, *Image compression via joint statistical characterization in the wavelet domain*, IEEE Trans. Image Process., 8 (1999), pp. 1688–1701.

[11] A.R. Calderbank and P.W. Shor, *Good quantum error-correcting codes exist*, Phys. Rev. A, 54 (1996), pp. 1098–1105.

[12] E.J. Candès and D.L. Donoho, *Recovering edges in ill-posed inverse problems: Optimality of curvelet frames*, Ann. Statist., 30 (2000), pp. 784–842.

[13] E.J. Candès and D.L. Donoho, *New tight frames of curvelets and optimal representations of objects with piecewise-$C^2$ singularities*, Comm. Pure Appl. Math., 57 (2002), pp. 219–266.

[14] E.J. Candès and J. Romberg, *Stable signal recovery from incomplete observations*, in Wavelet Applications in Signal and Image Processing XI, Proc. SPIE Conf. 5914, SPIE, Bellingham, WA, 2005.

[15] E.J. Candès, J. Romberg, and T. Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inform. Theory, 52 (2006), pp. 489–509.

[16] E. Candès, J. Romberg, and T. Tao, *Quantitative robust uncertainty principles and optimally sparse decompositions*, Found. Comput. Math., 6 (2006), pp. 227–254.

[17] E. Candès, J. Romberg, and T. Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Comm. Pure Appl. Math., 59 (2006), pp. 1207–1223.

[18] E.J. Candès and T. Tao, *Decoding by linear programming*, IEEE Trans. Inform. Theory, 51 (2005), pp. 4203–4215.

[19] S.G. Chang, B. Yu, and M. Vetterli, *Adaptive wavelet thresholding for image denoising and compression*, IEEE Trans. Image Process., 9 (2000), pp. 1532–1546.

[20] S.G. Chang, B. Yu, and M. Vetterli, *Wavelet thresholding for multiple noisy image copies*, IEEE Trans. Image Process., 9 (2000), pp. 1631–1635.

[21] S.G. Chang, B. Yu, and M. Vetterli, *Spatially adaptive wavelet thresholding with context modeling for image denoising*, IEEE Trans. Image Process., 9 (2000), pp. 1522–1530.

[22] V. Chandrasekaran, M. Wakin, D. Baron, and R. Baraniuk, *Surflets: A sparse representation for multidimensional functions containing smooth discontinuities*, in Proceedings of the IEEE Symposium on Information Theory, Chicago, IL, 2004, p. 563.

[23] S. Chen, S.A. Billings, and W. Luo, *Orthogonal least squares methods and their application to non-linear system identification*, Internat. J. Control, 50 (1989), pp. 1873–1896.

[24] S.S. Chen, D.L. Donoho, and M.A. Saunders, *Atomic decomposition by basis pursuit*, SIAM J. Sci. Comput., 20 (1998), pp. 33–61.

[25] S.S. Chen, D.L. Donoho, and M.A. Saunders, *Atomic decomposition by basis pursuit*, SIAM Rev., 43 (2001), pp. 129–159.

[26] A. Cohen, R.A. DeVore, P. Petrushev, and H. Xu, *Nonlinear approximation and the space $BV(\mathbb{R}^2)$*, Amer. J. Math., 121 (1999), pp. 587–628.

[27] R. Coifman and D.L. Donoho, *Translation-invariant denoising*, in Wavelets and Statistics, Lecture Notes in Statist. 103, Springer-Verlag, New York, 1995, pp. 120–150.

[28] R.R. Coifman, Y. Meyer, S. Quake, and M.V. Wickerhauser, *Signal processing and compression with wavelet packets*, in Progress in Wavelet Analysis and Applications (Toulouse, 1992), Frontières, 1993, pp. 77–93.

[29] R.R. Coifman and M.V. Wickerhauser, *Adapted waveform analysis as a tool for modeling, feature extraction, and denoising*, Optical Engrg., 33 (1994), pp. 2170–2174.

[30] C. Couvreur and Y. Bresler, *On the optimality of the backward greedy algorithm for the subset selection problem*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 797–808.

[31] C. Daniel and F.S. Wood, *Fitting Equations to Data: Computer Analysis of Multifactor Data*, 2nd ed., John Wiley and Sons, New York, 1980.

[32] I. Daubechies, M. Defrise, and C. De-Mol, *An iterative thresholding algorithm for linear inverse problems with a sparsity constraint*, Commun. Pure Appl. Math., 57 (2004), pp. 1413–1457.

[33] G. Davis, S. Mallat, and M. Avellaneda, *Adaptive greedy approximations*, J. Constructive Approx., 13 (1997), pp. 57–98.

[34] G. Davis, S. Mallat, and Z. Zhang, *Adaptive time-frequency decompositions*, Optical Engrg., 33 (1994), pp. 2183–2191.

[35] R.A. DeVore, B. Jawerth, and B.J. Lucier, *Image compression through wavelet transform coding*, IEEE Trans. Inform. Theory, 38 (1992), pp. 719–746.

[36] R.A. DeVore and V. Temlyakov, *Some remarks on greedy algorithms*, Adv. Comput. Math., 5 (1996), pp. 173–187.

[37] M.N. Do and M. Vetterli, *Rotation invariant texture characterization and retrieval using steerable wavelet-domain hidden Markov models*, IEEE Trans. Multimedia, 4 (2002), pp. 517–527.

[38] M.N. Do and M. Vetterli, *The finite ridgelet transform for image representation*, IEEE Trans. Image Process., 12 (2003), pp. 16–28.

[39] M.N. Do and M. Vetterli, *Framing pyramids*, IEEE Trans. Signal Process., 51 (2003), pp. 2329–2342.

[40] M.N. Do and M. Vetterli, *The contourlet transform: An efficient directional multiresolution image representation*, IEEE Trans. Image Process., 14 (2005), pp. 2091–2106.

[41] D.C. Dobson and F. Santosa, *Recovery of blocky images from noisy and blurred data*, SIAM J. Appl. Math., 56 (1996), pp. 1181–1198.

[42] D.L. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory, 52 (2006), pp. 1289–1306.

[43] D.L. Donoho, *De-noising by soft thresholding*, IEEE Trans. Inform. Theory, 41 (1995), pp. 613–627.

[44] D.L. Donoho, *For most large underdetermined systems of linear equations, the minimal $\ell_1$-norm solution is also the sparsest solution*, Commun. Pure Appl. Math., 59 (2006), pp. 797–829.

[45] D.L. Donoho, *For most large underdetermined systems of linear equations, the minimal $\ell_1$-norm near-solution approximates the sparsest near-solution*, Commun. Pure Appl. Math., 59 (2006), pp. 907–934.

[46] D.L. Donoho and M. Elad, *Optimally sparse representation in general (non-orthogonal) dictionaries via L1 minimization*, Proc. Natl. Acad. Sci., 100 (2003), pp. 2197–2202.

[47] D.L. Donoho and M. Elad, *On the stability of the basis pursuit in the presence of noise*, Signal Process., 86 (2006), pp. 511–532.

[48] D.L. Donoho, M. Elad, and V. Temlyakov, *Stable recovery of sparse overcomplete representations in the presence of noise*, IEEE Trans. Inform. Theory, 52 (2006), pp. 6–18.

[49] D.L. Donoho and X. Huo, *Uncertainty principles and ideal atomic decomposition*, IEEE Trans. Inform. Theory, 47 (1999), pp. 2845–2862.

[50] D.L. Donoho and I.M. Johnstone, *Ideal denoising in an orthonormal basis chosen from a library of bases*, C. R. Acad. Sci. Paris Sér. I Math., 319 (1994), pp. 1317–1322.

[51] D.L. Donoho and I.M. Johnstone, *Ideal spatial adaptation by wavelet shrinkage*, Biometrika, 81 (1994), pp. 425–455.

[52] D.L. Donoho and I.M. Johnstone, *Minimax estimation via wavelet shrinkage*, Ann. Statist., 26 (1998), pp. 879–921.

[53] D.L. Donoho, I.M. Johnstone, G. Kerkyacharian, and D. Picard, *Wavelet shrinkage—asymptopia*, J. Roy. Statist. Soc. Ser. B, 57 (1995), pp. 301–337.

[54] D.L. Donoho and P.B. Starck, *Uncertainty principles and signal recovery*, SIAM J. Appl. Math., 49 (1989), pp. 906–931.

[55] D.L. Donoho and V. Stodden, *When does non-negative matrix factorization give a correct decomposition into parts?*, in Proceedings of NIPS 2003, Adv. Neural Inform. Process. 16, MIT Press, Cambridge, MA, 2004.

[56] D.L. Donoho and J. Tanner, *Sparse nonnegative solutions of underdetermined linear equations by linear programming*, Proc. Natl. Acad. Sci., 102 (2005), pp. 9446–9451.

[57] D.L. Donoho and J. Tanner, *Neighborliness of randomly-projected simplices in high dimensions*, Proc. Natl. Acad. Sci., 102 (2005), pp. 9452–9457.

[58] D.L. Donoho and Y. Tsaig, *Extensions of compressed sensing*, Signal Process., 86 (2006), pp. 549–571.

[59] D.L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, *Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit*, IEEE Trans. Inform. Theory, submitted.

[60] B. Efron, T. Hastie, I.M. Johnstone, and R. Tibshirani, *Least angle regression*, Ann. Statist., 32 (2004), pp. 407–499.

[61] M. Elad, *Sparse representations are most likely to be the sparsest possible*, EURASIP J. Appl. Signal Process., 1 (2006), article 96247.

[62] M. Elad, *Why simple shrinkage is still relevant for redundant representations?*, IEEE Trans. Inform. Theory, 52 (2006), pp. 5559–5569.

[63] M. Elad and M. Aharon, *Image denoising via learned dictionaries and sparse representation*, in Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, New York, 2006, pp. 895–900.

[64] M. Elad and M. Aharon, *Image denoising via sparse and redundant representations over learned dictionaries*, IEEE Trans. Image Process., 15 (2006), pp. 3736–3745.

[65] M. Elad and A.M. Bruckstein, *A generalized uncertainty principle and sparse representation in pairs of bases*, IEEE Trans. Inform. Theory, 48 (2002), pp. 2558–2567.

[66] M. Elad, B. Matalon, and M. Zibulevsky, *Image denoising with shrinkage and redundant representations*, in Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, New York, 2006, pp. 1924–1931.

[67] M. Elad, B. Matalon, and M. Zibulevsky, *Coordinate and subspace optimization methods for linear least squares with non-quadratic regularization*, Appl. Comput. Harmon. Anal., 23 (2007), pp. 346–367.

[68] M. Elad, J.-L. Starck, P. Querre, and D.L. Donoho, *Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)*, Appl. Comput. Harmon. Anal., 19 (2005), pp. 340–358.

[69] K. Engan, S.O. Aase, and J.H. Husoy, *Multi-frame compression: Theory and design*, Signal Process., 80 (2000), pp. 2121–2140.

[70] R. Eslami and H. Radha, *The contourlet transform for image de-noising using cycle spinning*, in Proceedings of Asilomar Conference on Signals, Systems, and Computers, 2003, pp. 1982–1986.

[71] R. Eslami and H. Radha, *Translation-invariant contourlet transform and its application to image denoising*, IEEE Trans. Image Process., 15 (2006), pp. 3362–3374.

[72] M.J. Fadili and J.-L. Starck, *Sparse representation-based image deconvolution by iterative thresholding*, Astronomical Data Analysis ADA'06, Marseilles, France, 2006.

[73] A. Feuer and A. Nemirovsky, *On sparse representation in pairs of bases*, IEEE Trans. Inform. Theory, 49 (2002), pp. 1579–1581.

[74] M.A. Figueiredo, J.M. Bioucas-Dias, and R.D. Nowak, *Majorization-minimization algorithms for wavelet-based image restoration*, IEEE Trans. Image Process., 16 (2007), pp. 2980–2991.

[75] M.A. Figueiredo and R.D. Nowak, *A bound optimization approach to wavelet-based image deconvolution*, in Proceedings of the IEEE International Conference on Image Processing (ICIP 2005), Genoa, Italy, 2005, pp. 782–785.

[76] M.A. Figueiredo and R.D. Nowak, *An EM algorithm for wavelet-based image restoration*, IEEE Trans. Image Process., 12 (2003), pp. 906–916.

[77] A.K. Fletcher, S. Rangan, V.K. Goyal, and K. Ramchandran, *Analysis of denoising by sparse approximation with random frame asymptotics*, in Proceedings of the IEEE International Symposium on Information Theory, 2005, pp. 1706–1710.

[78] A.K. Fletcher, S. Rangan, V.K. Goyal, and K. Ramchandran, *Denoising by sparse approximation: Error bounds based on rate-distortion theory*, EURASIP J. Appl. Signal Process., 1 (2006), article 26318.

[79] J.J. Fuchs, *On sparse representations in arbitrary redundant bases*, IEEE Trans. Inform. Theory, 50 (2004), pp. 1341–1344.

[80] J.J. Fuchs, *Recovery of exact sparse representations in the presence of bounded noise*, IEEE Trans. Inform. Theory, 51 (2005), pp. 3601–3608.

[81] A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic, Dordrecht, The Netherlands, 1992.

[82] A.C. Gilbert, S. Muthukrishnan, and M.J. Strauss, *Approximation of functions over redundant dictionaries using coherence*, in Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms, ACM, New York, SIAM, Philadelphia, 2003, pp. 243–252.

[83] G.H. Golub and C.F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[84] I.F. Gorodnitsky and B.D. Rao, *Sparse signal reconstruction from limited data using FOCUSS: A re-weighted norm minimization algorithm*, IEEE Trans. Signal Process., 45 (1997), pp. 600–616.

[85] R. Gribonval, R. Figueras, and P. Vandergheynst, *A simple test to check the optimality of a sparse signal approximation*, Signal Process., 86 (2006), pp. 496–510.

[86] R. Gribonval and M. Nielsen, *Sparse decompositions in unions of bases*, IEEE Trans. Inform. Theory, 49 (2003), pp. 3320–3325.

[87] R. Gribonval and P. Vandergheynst, *On the exponential convergence of matching pursuits in quasi-incoherent dictionaries*, IEEE Trans. Inform. Theory, 52 (2006), pp. 255–261.

[88] O.G. Guleryuz, *Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising—Part* I: *Theory*, IEEE Trans. Image Process., 15 (2006), pp. 539–554.

[89] O.G. Guleryuz, *Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising—Part* II: *Adaptive algorithms*, IEEE Trans. Image Process., 15 (2006), pp. 555–571.

[90] T. Hastie, R. Tibshirani, and J.H. Friedman, *Elements of Statistical Learning*, Springer-Verlag, New York, 2001.

[91] R.A. Horn and C.R. Johnson, *Matrix Analysis*, Cambridge University Press, New York, 1985.

[92] P.O. Hoyer, *Non-negative matrix factorization with sparseness constraints*, J. Machine Learning Res., 5 (2004), pp. 1457–1469.

[93] X. Huo, *Sparse Image representation via Combined Transforms*, Ph.D. thesis, Stanford, 1999.

[94] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, Wiley Interscience, New York, 2001.

[95] A.K. Jain, *Fundamentals of Digital Image Processing*, Prentice–Hall, Englewood Cliffs, NJ, 1989.

[96] M. Jansen, *Noise Reduction by Wavelet Thresholding*, Springer-Verlag, New York, 2001.

[97] L.A. Karlovitz, *Construction of nearest points in the $\ell_p$, $p$ even and $\ell_\infty$ norms*, J. Approx. Theory, 3 (1970), pp. 123–127.

[98] B. Kashin, *The widths of certain finite-dimensional sets and classes of smooth functions*, Izv. Akad. Nauk SSSR Ser. Mat., 41 (1977), pp. 334–351.

[99] E. Kidron, Y.Y. Schechner, and M. Elad, *Cross-modality localization via sparsity*, IEEE Trans. Signal Process., 55 (2007), pp. 1390–1404.

[100] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, *A method for large-scale $\ell_1$-regularized least squares problems with applications in signal processing and statistics*, IEEE J. Sel. Topics Signal Proc., 1 (2007), pp. 606–617.

[101] K. Kreutz-Delgado, J.F. Murray, B.D. Rao, K. Engan, T.-W, Lee, and T.J. Sejnowski, *Dictionary learning algorithms for sparse representation*, Neural Comput., 15 (2003), pp. 349–396.

[102] J.B. Kruskal, *Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics*, Linear Algebra Appl., 18 (1977), pp. 95–138.

[103] M. Lang, H. Guo, and J.E. Odegard, *Noise reduction using undecimated discrete wavelet transform*, IEEE Signal Process. Lett., 3 (1996), pp. 10–12.

[104] D. Lee and H. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature, 401 (1999), pp. 788–791.

[105] E. Le Pennec and S. Mallat, *Sparse geometric image representation with bandelets*, IEEE Trans. Image Process., 14 (2005), pp. 423–438.

[106] S. Lesage, R. Gribonval, F. Bimbot, and L. Benaroya, *Learning unions of orthonormal bases with thresholded singular value decomposition*, in Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing (ICASSP 2005), 2005, pp. 293–296.

[107] M.S. Lewicki and B.A. Olshausen, *A probabilistic framework for the adaptation and comparison of image codes*, J. Opt. Soc. Amer. A, 16 (1999), pp. 1587–1601.

[108] M.S. Lewicki and T.J. Sejnowski, *Learning overcomplete representations*, Neural Comput., 12 (2000), pp. 337–365.

[109] Y. Li, A. Cichocki, and S.-i. Amari, *Analysis of sparse representation and blind source separation*, Neural Comput., 16 (2004), pp. 1193–1234.

[110] X. Liu and N.D. Sidiropoulos, *Cramer–Rao lower bounds for low-rank decomposition of multidimensional arrays*, IEEE Trans. Signal Process., 49 (2001), pp. 2074–2086.

[111] M. Lustig, D.L. Donoho, and J.M. Pauly, *Sparse MRI: The application of compressed sensing for rapid MR imaging*, Magnetic Resonance in Medicine, 58 (2007), pp. 1182–1195.

[112] M. Lustig, J.M. Santos, D.L. Donoho, and J.M. Pauly, *k-t SPARSE: High frame rate dynamic MRI exploiting spatio-temporal sparsity*, in Proceedings of the 13th Annual Meeting of ISMRM, Seattle, 2006.

[113] J. Mairal, M. Elad, and G. Sapiro, *Sparse representation for color image restoration*, IEEE Trans. Image Process., 17 (2008), pp. 53–69.

[114] D.M. Malioutov, M. Cetin, and A.S. Willsky, *Optimal sparse representations in general overcomplete bases*, in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Montreal, Canada, 2004, pp. 793–796.

[115] D.M. Malioutov, M. Cetin, and A.S. Willsky, *Sparse signal reconstruction perspective for source localization with sensor arrays*, IEEE Trans. Signal Process., 53 (2005), pp. 3010–3022.

[116] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, San Diego, CA, 1998.

[117] S. Mallat and E. Le Pennec, *Bandelet image approximation and compression*, Multiscale Model. Simul., 4 (2005), pp. 992–1039.

[118] S. Mallat and Z. Zhang, *Matching pursuits with time-frequency dictionaries*, IEEE Trans. Signal Process., 41 (1993), pp. 3397–3415.

[119] S. Mendelson, A. Pajor, and N. Tomczak-Jaegermann, *Reconstruction and subGaussian processes*, C. R. Math. Acad. Sci. Paris, 340 (2005), pp. 885–888.

[120] S. Mendelson, A. Pajor, and N. Tomczak-Jaegermann, *Uniform uncertainty principle for Bernoulli and subGaussian ensembles*, Constructive Approx., to appear.

[121] F.G. Meyer, A. Averbuch, and J.O. Stromberg, *Fast adaptive wavelet packet image compression*, IEEE Trans. Image Process., 9 (2000), pp. 792–800.

[122] F.G. Meyer, A.Z. Averbuch, and R.R. Coifman, *Multilayered image representation: Application to image compression*, IEEE Trans. Image Process., 11 (2002), pp. 1072–1080.

[123] F.G. Meyer and R.R. Coifman, *Brushlets: A tool for directional image analysis and image compression*, Appl. Comput. Harmon. Anal., 4 (1997), pp. 147–187.

[124] P. Moulin and J. Liu, *Analysis of multiresolution image denoising schemes using generalized Gaussian and complexity priors*, IEEE Trans. Inform. Theory, 45 (1999), pp. 909–919.

[125] B.K. Natarajan, *Sparse approximate solutions to linear systems*, SIAM J. Comput., 24 (1995), pp. 227–234.

[126] B.A. Olshausen and B.J. Field, *Emergence of simple-cell receptive field properties by learning a sparse code for natural images*, Nature, 381 (1997), pp. 607–609.

[127] B.A. Olshausen and B.J. Field, *Sparse coding with an overcomplete basis set: A strategy employed by V1?*, Vision Res., 37 (1997), pp. 3311–3325.

[128] B.A. Olshausen and D.J. Field, *Natural image statistics and efficient coding*, Network—Computation in Neural Systems, 7 (1996), pp. 333–339.

[129] M.R. Osborne, B. Presnell, and B.A. Turlach, *A new approach to variable selection in least squares problems*, IMA J. Numer. Anal., 20 (2000), pp. 389–403.

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

[131] W.W. Peterson and E.J. Weldon, Jr., *Error-Correcting Codes*, 2nd ed., MIT Press, Cambridge, MA, 1972.

[132] G. Pisier, *The Volume of Convex Bodies and Banach Space Geometry*, Cambridge University Press, Cambridge, UK, 1989.

[133] M.D. Plumbley, *Geometry and homotopy for L1 sparse representations*, in Proceedings of the Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS'05), Rennes, France, 2005.

[134] M.D. Plumbley, *Polar Polytopes and Recovery of Sparse Representations*, preprint, 2005; available online from http://arxiv.org/abs/cs/0510032v1.

[135] M.D. Plumbley, *Recovery of sparse representations by polytope faces pursuit*, in Proceedings of the 6th International Conference on Independent Component Analysis and Blind Source Separation (ICA 2006), Charleston, SC, Lecture Notes in Comput. Sci. 3889, Springer-Verlag, Berlin, 2006, pp. 206–213.

[136] J. Portilla, V. Strela, M.J. Wainwright, and E.P. Simoncelli, *Image denoising using scale mixtures of Gaussians in the wavelet domain*, IEEE Trans. Image Process., 12 (2003), pp. 1338–1351.

[137] M. Protter and M. Elad, *Image sequence denoising via sparse and redundant representations*, IEEE Trans. Image Process., to appear.

[138] B.D. Rao, K. Engan, S.F. Cotter, J. Palmer, and K. Kreutz-Delgado, *Subset selection in noise based on diversity measure minimization*, IEEE Trans. Signal Process., 51 (2003), pp. 760–770.

[139] B.D. Rao and K. Kreutz-Delgado, *An affine scaling methodology for best basis selection*, IEEE Trans. Signal Process., 47 (1999), pp. 187–200.

[140] M. Rudelson and R. Vershynin, *Geometric Approach to Error-Correcting Codes and Reconstruction of Signals*, Technical Report, Department of Mathematics, University of California, Davis, CA, 2005.

[141] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.

[142] F. Santosa and W.W. Symes, *Linear inversion of band-limited reflection seismograms*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 1307–1330.

[143] S. Sardy, A.G. Bruce, and P. Tseng, *Block coordinate relaxation methods for nonparametric signal denoising with wavelet dictionaries*, J. Comput. Graphical Statist., 9 (2000), pp. 361–379.

[144] E.P. Simoncelli and E.H. Adelson, *Noise removal via Bayesian wavelet coring*, in Proceedings of the International Conference on Image Processing, Lausanne, Switzerland, 1996, pp. 379–382.

[145] E.P. Simoncelli, W.T. Freeman, E.H. Adelson, and D.J. Heeger, *Shiftable multi-scale transforms*, IEEE Trans. Inform. Theory, 38 (1992), pp. 587–607.

[146] J.-L. Starck, E.J. Candès, and D.L. Donoho, *The curvelet transform for image denoising*, IEEE Trans. Image Process., 11 (2002), pp. 670–684.

[147] J.-L. Starck, M. Elad, and D.L. Donoho, *Image decomposition via the combination of sparse representations and a variational approach*, IEEE Trans. Image Process., 14 (2005), pp. 1570–1582.

[148] J.-L. Starck, M. Elad, and D.L. Donoho, *Redundant multiscale transforms and their application for morphological component separation*, Adv. Imaging Electron Phys., 132 (2004), pp. 287–348.

[149] J.-L. Starck, M.J. Fadili, and F. Murtagh, *The undecimated wavelet decomposition and its reconstruction*, IEEE Trans. Image Process., 16 (2007), pp. 297–309.

[150] T. Strohmer and R. W. Heath, *Grassmannian frames with applications to coding and communication*, Appl. Comput. Harmon. Anal., 14 (2004), pp. 257–275.

[151] S. Szarek, *Condition number of random matrices*, J. Complexity, 7 (1991), pp. 131–149.

[152] S. Szarek, *Spaces with large distance to $\ell^\infty$ and random matrices*, Amer. J. Math., 112 (1990), pp. 899–942.

[153] D.S. Taubman and M.W. Marcellin, *JPEG* 2000*: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic, Norwell, MA, 2001.

[154] V.N. Temlyakov, *Greedy algorithms and m-term approximation*, J. Approx. Theory, 98 (1999), pp. 117–145.

[155] V.N. Temlyakov, *Weak greedy algorithms*, Adv. Comput. Math., 5 (2000), pp. 173–187.

[156] J.A. Tropp, *Greed is good: Algorithmic results for sparse approximation*, IEEE Trans. Inform. Theory, 50 (2004), pp. 2231–2242.

[157] J.A. Tropp, *Just relax: Convex programming methods for subset selection and sparse approximation*, IEEE Trans. Inform. Theory, 52 (2006), pp. 1030–1051.

[158] J.A. Tropp and A.A. Gilbert, *Signal recovery from random measurements via orthogonal matching pursuit*, IEEE Trans. Inform. Theory. 53 (2007), pp. 4655–4666.

[159] J.A. Tropp, A.C. Gilbert, S. Muthukrishnan, and M.J. Strauss, *Improved sparse approximation over quasi-incoherent dictionaries*, in Proceedings of the IEEE International Conference on Image Processing, Barcelona, 2003, pp. 37–40.

[160] Y. Tsaig, *Sparse Solution of Underdetermined Linear Systems: Algorithms and Applications*, Ph.D. thesis, Stanford, CA, 2007.

[161] Y. Tsaig and D.L. Donoho, *Breakdown of equivalence between the minimal $L_1$-norm solution and the sparsest solution*, Signal Process., 86 (2006), pp. 533–548.

[162] H. Wintney, *On the abstract properties of linear dependence*, Amer. J. Math., 57 (1935), pp. 509–533.

[163] B. Wohlberg, *Noise sensitivity of sparse signal representations: Reconstruction error bounds for the inverse problem*, IEEE Trans. Signal Process., 51 (2003), pp. 3053–3060.

[164] M. Zibulevsky and B.A. Pearlmutter, *Blind source separation by sparse decomposition in a signal dictionary*, Neural Comput., 13 (2001), pp. 863–882.