

SPARSE DECOMPOSITION OF TWO DIMENSIONAL SIGNALS

Aboozar Ghaffari, Massoud Babaie-Zadeh*

Electrical Engineering Department, Sharif
University of Technology, Tehran, Iran

Christian Jutten

GIPSA-LAB, Grenoble, France

ABSTRACT

In this paper, we consider sparse decomposition (SD) of two-dimensional (2D) signals on overcomplete dictionaries with separable atoms. Although, this problem can be solved by converting it to the SD of one-dimensional (1D) signals, this approach requires a tremendous amount of memory and computational cost. Moreover, the uniqueness constraint obtained by this approach is too restricted. Then in the paper, we present an algorithm to be used directly for sparse decomposition of 2D signals on dictionaries with separable atoms. Moreover, we will state another uniqueness constraint for this class of decomposition. Our algorithm is obtained by modifying the Smoothed L0 (SL0) algorithm, and hence we call it two-dimensional SL0 (2D-SL0).

Index Terms— Sparse Representation, Sparse Decomposition, Compressive Sensing, Image Coding, Sparse Coding.

1. INTRODUCTION

In signal (or atomic [1]) decomposition, a signal vector $\mathbf{x} \in \mathbb{R}^n$ is to be decomposed as a linear combination of m basic signals $\varphi_i \in \mathbb{R}^n, 1 \leq i \leq m$, that is, $\mathbf{x} = s_1\varphi_1 + \dots + s_m\varphi_m = \Phi\mathbf{s}$, where $\Phi = [\varphi_1, \dots, \varphi_m]$ and $\mathbf{s} = (s_1, \dots, s_m)^T$. After [2], the signals $\varphi_i, i = 1, \dots, m$ are called ‘atoms’, and they collectively form the ‘dictionary’ Φ . In classical signal decomposition (e.g. in Discrete Fourier Transform), the number of atoms (m) is equal to the length of signals (n), and the decomposition is then unique. However, where the dictionary is overcomplete, ie. where $m > n$, the decomposition is not unique, and the goal of *sparse decomposition* (SD) is then to search for a decomposition in which as few as possible atoms are present in the decomposition. Mathematically, this is equivalent to finding the sparsest solution of the Underdetermined System of Linear Equations (USLE) $\Phi\mathbf{s} = \mathbf{x}$. This problem has recently attracted a lot of interest in signal processing and statistics, because of its application in many different areas, for example, in under-

determined Sparse Component Analysis (SCA) [3] and compressed sensing [4].

Sparse decomposition of the signal $\mathbf{x} \in \mathbb{R}^n$ over the redundant dictionary $\Phi \in \mathbb{R}^{n \times m}, m > n$, can be stated as:

$$(P_0) : \text{Minimize } \|\mathbf{s}\|_0 \text{ subject to } \Phi\mathbf{s} = \mathbf{x} \quad (1)$$

where $\|\mathbf{s}\|_0$ stands for the ℓ^0 norm of \mathbf{s} , that is, the number of non-zero elements of \mathbf{s} . Two fundamental questions are then:

1. (Uniqueness) Under which conditions the solution of this problem is unique?
2. (Decomposition algorithm) How to find this solution reliably and efficiently in practice?

To answer the first question, let *spark* of a matrix [5] stand for the minimal number of its columns which are linearly dependent¹. Then, it can be shown [6, 5, 7] that if the USLE $\Phi\mathbf{s} = \mathbf{x}$ has a solution for which $\|\mathbf{s}\|_0 < \text{spark}(\Phi)/2$, it is the unique sparsest solution. As answers to the second question, several methods have been developed in the recent years. Just as some examples, we mention here Matching Pursuit (MP) [2], Basis Pursuit (BP) [1], FOCUSS [6] and Smoothed ℓ^0 (SL0) [8].

Now consider decomposition of two-dimensional (2D) signals: $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$ is to be decomposed as linear combination of atoms $\Phi_{ij}, 1 \leq i \leq m_1, 1 \leq j \leq m_2$, that is $\mathbf{X} = \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} s_{ij} \Phi_{ij}$. Letting $\mathbf{x} = \text{vec}(\mathbf{X})$ and $\varphi_{ij} = \text{vec}(\Phi_{ij})$, where ‘vec’ of a matrix stands for the vector obtained by stacking its columns, this 2D problem will be converted to the previous 1D problem.

However, the atoms usually used in 2D signal transformation and representation (eg. 2D Fourier atoms) [9] are separable, that is, there are vectors \mathbf{a}_i and \mathbf{b}_j such that $\Phi_{ij} = \mathbf{a}_i \mathbf{b}_j^T, 1 \leq i \leq m_1, 1 \leq j \leq m_2$. This kind of atoms result in 2D decompositions (and transformations) of the form:

$$\mathbf{X}_{n_1 \times n_2} = \mathbf{A} \mathbf{S}_{m_1 \times m_2} \mathbf{B}^T \quad (2)$$

¹Note that for an ‘all-columns-independent’ matrix (a matrix for which all columns are independent) the *spark* has not been defined. One may propose to define the spark of such a matrix as the number of its columns plus one. We do not apply such a definition in this paper, otherwise some results (e.g. Lemma 3) would not be correct. In our point of view, the existence of the spark of a matrix implies also that it has some dependent columns.

*This work has been partially supported by Iran NSF (INSF) under contract number 86/994, by Iran Telecommunications Research Center (ITRC), and also by ISMO and French embassy in Iran in the framework of a GundiShapour collaboration program.

where $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_{m_1}]$ and $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_{m_2}]$. For the complete case, i.e. where $n_1 = m_1$ and $n_2 = m_2$, the above decomposition is unique. However, for the overcomplete case, i.e. where $n_1 < m_1$ and $n_2 < m_2$, the representation is not unique, and in Sparse Decomposition (SD), we would like to find the \mathbf{S} with as much zero elements as possible.

A trivial approach for solving this 2D sparse decomposition problem is to write:

$$\mathbf{X} = \mathbf{A}\mathbf{S}\mathbf{B}^T \Leftrightarrow \Phi \mathbf{s} = \mathbf{x} \quad (3)$$

where $\mathbf{x} = \text{vec}(\mathbf{X})$, $\mathbf{s} = \text{vec}(\mathbf{S})$, and $\Phi = \mathbf{B} \otimes \mathbf{A}$ is the Kronecker product [10] of \mathbf{A} and \mathbf{B} ; and then to use algorithms developed for 1D sparse decomposition. The problem is, however, that the resulting USLE is too large to be solved practically. For example, for SD of a small image $\mathbf{X} \in \mathbb{R}^{40 \times 50}$ on a dictionary with $\mathbf{A} \in \mathbb{R}^{40 \times 100}$ and $\mathbf{B} \in \mathbb{R}^{50 \times 100}$, this approach results in Φ with dimensions 2000×10000 , and SD over such a large dictionary requires a very high computational load and memory.

The goal of this paper is then to obtain an algorithm to solve directly the sparse decomposition in (2). More precisely, we will modify the SL0 algorithm to directly work for (2). Moreover, we will consider the uniqueness of the 2D sparse decomposition, and we will see that the uniqueness condition given by the trivial approach (3) is very restricted, and less restrictive conditions on \mathbf{A} and \mathbf{B} may be possible.

The paper is organized as follows. Firstly, Section 2 discusses the uniqueness of SD in model (2). Our 2D sparse decomposition algorithm will then be stated in Section 3. Finally, Section 4 provides some experimental results.

2. UNIQUENESS

Let first consider the uniqueness condition obtained from (3). The uniqueness of the sparsest solution of USLE $\Phi \mathbf{s} = \mathbf{x}$ imposes an upper bound on $\|\mathbf{s}\|_0$, which is $\text{spark}(\Phi)/2 = \text{spark}(\mathbf{B} \otimes \mathbf{A})/2$. To obtain an upper bound on $\|\mathbf{s}\|_0$ in terms of the matrices \mathbf{A} and \mathbf{B} , we obtain an upper bound for $\text{spark}(\mathbf{B} \otimes \mathbf{A})$ in terms of $\text{spark}(\mathbf{A})$ and $\text{spark}(\mathbf{B})$. To do this, let state the following lemmas. The first two lemmas follow easily from the definition of spark.

Lemma 1. For arbitrary vector \mathbf{v} and matrix $\mathbf{U} \in \mathbb{R}^{n \times m}$, we have $\text{spark}(\mathbf{v} \otimes \mathbf{U}) = \text{spark}(\mathbf{U})$.

Note also that the above lemma implies that the matrix $\mathbf{v} \otimes \mathbf{U}$, although can be tall, has some dependent columns (it has a well-defined spark, see the footnote of the previous page).

Lemma 2. A change in arrangement of rows or columns of a matrix does not change its spark.

Lemma 3. Let $\mathbf{M} = [\mathbf{M}_1, \dots, \mathbf{M}_n]$ be a partitioning of the matrix \mathbf{M} , where sub-matrices \mathbf{M}_i have well-defined sparks. Then:

$$\forall i: \text{spark}(\mathbf{M}) \leq \text{spark}(\mathbf{M}_i) \quad (4)$$

and hence $\text{spark}(\mathbf{M}) \leq \min\{\text{spark}(\mathbf{M}_1), \dots, \text{spark}(\mathbf{M}_n)\}$.

Proof. Note that the assumption that \mathbf{M}_i has a well-defined spark means that there are some dependent columns in it (see the footnote of the previous page). Now, let $\text{spark}(\mathbf{M}_i) = l_i$. This means that there exist already $l_i + 1$ dependent columns in \mathbf{M}_i . Hence the minimum number of dependent columns in \mathbf{M} is less than or equal to $l_i + 1$, which gives (4). \square

Theorem 1. Let $\Phi = \mathbf{B} \otimes \mathbf{A}$. Then:

$$\text{spark}(\Phi) \leq \min\{\text{spark}(\mathbf{A}), \text{spark}(\mathbf{B})\} \quad (5)$$

Proof. Let $\Phi_k \triangleq \mathbf{b}_k \otimes \mathbf{A}$, $k = 1, \dots, m_2$. Then $\Phi = [\Phi_1, \dots, \Phi_{m_2}]$. From Lemma 1, $\forall k, \text{spark}(\Phi_k) = \text{spark}(\mathbf{A})$, and hence Lemma 3 implies $\text{spark}(\Phi) \leq \text{spark}(\mathbf{A})$.

Now let Ψ_i , $i = 1, \dots, m_1$ be a matrix formed by collecting the i -th column of all matrices Φ_k , $k = 1, \dots, m_2$. Since the i -th column of Φ_k is, in fact, $\mathbf{b}_k \otimes \mathbf{a}_i$, we have $\Psi_i = [\mathbf{b}_1 \otimes \mathbf{a}_i, \mathbf{b}_2 \otimes \mathbf{a}_i, \dots, \mathbf{b}_{m_2} \otimes \mathbf{a}_i]$. It is easy then to see that a re-arrangement of the rows of Ψ_i gives $\mathbf{a}_i \otimes \mathbf{B}$, and hence Lemmas 2 and 1 imply $\text{spark}(\Psi_i) = \text{spark}(\mathbf{B})$. Moreover, the matrix $[\Psi_1, \dots, \Psi_{m_1}]$ is just a re-arrangement of the columns of Φ , and hence the above lemmas imply that $\text{spark}(\Phi) = \text{spark}([\Psi_1, \dots, \Psi_{m_1}]) \leq \text{spark}(\Psi_i) = \text{spark}(\mathbf{B})$. \square

From the above theorem, for \mathbf{S} being the unique sparsest solution of (2), it is necessary (and not even sufficient) that $\|\mathbf{S}\|_0 < \frac{1}{2} \min\{\text{spark}(\mathbf{A}), \text{spark}(\mathbf{B})\}$. This requires a highly sparse \mathbf{S} compared to its dimensions, that is, just a very very small percentage of the entries of \mathbf{S} should be non-zero. In stating this result, no assumption about the distribution of these non-zero entries in \mathbf{S} has been made (it includes for example the case all non-zero elements of \mathbf{S} are concentrated in just one row).

However, if the non-zero entries of \mathbf{S} are distributed in the whole matrix, i.e. if all of its rows and columns are relatively sparse, $\|\mathbf{S}\|_0$ can be highly larger than what implied by (5), without violating the uniqueness. To state a uniqueness theorem based on this idea, we first state the following lemma.

Lemma 4. Let $\mathbf{A} \in \mathbb{R}^{n_1 \times m_1}$ and $\mathbf{B} \in \mathbb{R}^{n_2 \times m_2}$ be wide matrices, and consider the underdetermined system of linear equations $\mathbf{A}\mathbf{S}\mathbf{B}^T = \mathbf{0}$. Suppose it is known that the non-zero elements of \mathbf{S} have been distributed in at most $(\text{spark}(\mathbf{A}) - 1)$ rows and at most $(\text{spark}(\mathbf{B}) - 1)$ columns of \mathbf{S} . Then $\mathbf{S} = \mathbf{0}$.

Proof. Without loss of generality, assume \mathbf{S} has the following form:

$$\mathbf{S} = \begin{pmatrix} \hat{\mathbf{S}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

where $\hat{\mathbf{S}}$ is a $(\text{spark}(\mathbf{A}) - 1)$ by $(\text{spark}(\mathbf{B}) - 1)$ unknown matrix. Consequently, $\mathbf{A}\mathbf{S}\mathbf{B}^T = \mathbf{0}$ can be rewritten as:

$$[\mathbf{A}_1, \mathbf{A}_2] \begin{pmatrix} \hat{\mathbf{S}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} [\mathbf{B}_1, \mathbf{B}_2]^T = \mathbf{0} \Rightarrow \mathbf{A}_1 \hat{\mathbf{S}} \mathbf{B}_1^T = \mathbf{0} \quad (6)$$

where \mathbf{A}_1 and \mathbf{B}_1 are tall and full rank matrices (from the definition of spark), and hence $\mathbf{A}_1^T \mathbf{A}_1$ and $\mathbf{B}_1^T \mathbf{B}_1$ are invertible. Multiplying the above equation from left by $(\mathbf{A}_1^T \mathbf{A}_1)^{-1} \mathbf{A}_1^T$ and from right by $\mathbf{B}_1 (\mathbf{B}_1^T \mathbf{B}_1)^{-1}$ proves the lemma. \square

Theorem 2 (2D Uniqueness Theorem). *If the system of linear equation $\mathbf{X} = \mathbf{A}\mathbf{S}\mathbf{B}^T$ has a solution \mathbf{S} for which all of the nonzero entries are distributed in strictly less than $\text{spark}(\mathbf{A})/2$ rows and $\text{spark}(\mathbf{B})/2$ columns, then it is the unique sparsest solution.*

Proof. Suppose that \mathbf{S}_1 and \mathbf{S}_2 are both solutions of $\mathbf{X} = \mathbf{A}\mathbf{S}\mathbf{B}^T$, satisfying the above condition. Then $\mathbf{A}(\mathbf{S}_1 - \mathbf{S}_2)\mathbf{B}^T = \mathbf{0}$. Now $\mathbf{S}_1 - \mathbf{S}_2$ has the condition of Lemma 4, and hence $\mathbf{S}_1 - \mathbf{S}_2 = \mathbf{0}$. \square

Some necessary (but not sufficient) conditions for \mathbf{S} satisfying the condition of the above theorem are:

- $\|\mathbf{S}\|_0 < \text{spark}(\mathbf{A})\text{spark}(\mathbf{B})/4$.
- (ℓ^0 norm of each column of \mathbf{S}) $< \text{spark}(\mathbf{A})/2$.
- (ℓ^0 norm of each row of \mathbf{S}) $< \text{spark}(\mathbf{B})/2$.

3. FINDING THE SPARSE SOLUTION

Consider now the problem ‘‘how to practically find the sparsest solution to (2)’’? Although it is always possible to do the conversion in (3) to convert this 2D problem to a 1D problem, this approach results in huge matrices, and requires a tremendous amount of memory and computational cost. Hence in this section, we obtain an algorithm based on a modification of SL0 [8], which directly works with (2).

In the 1D case $\Phi\mathbf{s} = \mathbf{x}$, the idea of SL0 is to maximize $F_\sigma(\mathbf{s}) \triangleq \sum_i \exp(-s_i^2/2\sigma^2)$ for a small σ , subject to $\Phi\mathbf{s} = \mathbf{x}$. To escape from trapping into local maxima, it uses a decreasing sequence of σ , and the maximizer of F_σ is used as a starting point to search the maximizer of F_σ for the next (smaller) σ . To maximize F_σ for a fixed σ subject to $\Phi\mathbf{s} = \mathbf{x}$, it uses a steepest ascent approach: each iteration composed of an unconstrained maximization step ($\mathbf{s}' \leftarrow \mathbf{s} + \mu\nabla F_\sigma$), followed by projection to the feasible set $\mathcal{S} = \{\mathbf{s} \mid \Phi\mathbf{s} = \mathbf{x}\}$:

$$\mathbf{s} = \underset{\mathbf{s}}{\text{argmin}} \|\mathbf{s} - \mathbf{s}'\| \quad \text{s.t.} \quad \mathbf{s} \in \mathcal{S} \quad (7)$$

$$= \mathbf{s}' - \Phi^\dagger(\Phi\mathbf{s}' - \mathbf{x}), \quad (8)$$

where $\Phi^\dagger = \Phi^T(\Phi\Phi^T)^{-1}$ is the pseudoinverse of Φ . Moreover, the algorithm is initialized by the minimum ℓ^2 norm solution of $\Phi\mathbf{s} = \mathbf{x}$, that is, $\Phi^\dagger\mathbf{x}$. This is justified [8] because this solution corresponds to the maximizer of F_σ for $\sigma \rightarrow \infty$.

For modifying SL0 to directly work with (2), we should modify the projection step (7) by replacing \mathbf{s} with \mathbf{S} and \mathcal{S} with $\{\mathbf{S} \mid \mathbf{A}\mathbf{S}\mathbf{B}^T = \mathbf{X}\}$. Moreover, we should modify the initialization. This is given in the following theorems.

Theorem 3 (Projection). *The projection of a matrix \mathbf{S}' onto the feasible set $\{\mathbf{S} \mid \mathbf{A}\mathbf{S}\mathbf{B}^T = \mathbf{X}\}$ is given by:*

$$\mathbf{S} = \mathbf{S}' - \mathbf{A}^\dagger(\mathbf{A}\mathbf{S}'\mathbf{B}^T - \mathbf{X})(\mathbf{B}^\dagger)^T \quad (9)$$

where $\mathbf{A}^\dagger = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$ and $\mathbf{B}^\dagger = \mathbf{B}^T(\mathbf{B}\mathbf{B}^T)^{-1}$ are the pseudoinverses of \mathbf{A} and \mathbf{B} , respectively.

Proof. Applying $\text{vec}()$ on both sides of (9), we conclude that (9) is equivalent to:

$$\mathbf{s} = \mathbf{s}' - \text{vec}\{\mathbf{A}^\dagger(\mathbf{A}\mathbf{S}'\mathbf{B}^T - \mathbf{X})(\mathbf{B}^\dagger)^T\} \quad (10)$$

where $\mathbf{s} = \text{vec}(\mathbf{S})$ and $\mathbf{s}' = \text{vec}(\mathbf{S}')$. Similar to (3), this equation can be written as $\mathbf{s} = \mathbf{s}' - (\mathbf{B}^\dagger \otimes \mathbf{A}^\dagger)\text{vec}(\Delta)$, where $\Delta = \mathbf{A}\mathbf{S}'\mathbf{B}^T - \mathbf{X}$. Using (3) again, we have $\text{vec}(\Delta) = \Phi\mathbf{s} - \mathbf{x}$, where $\Phi = \mathbf{B} \otimes \mathbf{A}$ and $\mathbf{x} = \text{vec}(\mathbf{X})$. Moreover, from the properties of the Kronecker product [10] we have $\mathbf{B}^\dagger \otimes \mathbf{A}^\dagger = (\mathbf{B} \otimes \mathbf{A})^\dagger = \Phi^\dagger$. Replacing these in the above equation, (10) leads to (8). In other words, (9) is equivalent to the projection step for the converted system (3). \square

Theorem 4 (Initialization). *The minimum ℓ^2 norm solution of $\mathbf{A}\mathbf{S}\mathbf{B}^T = \mathbf{X}$ is*

$$\hat{\mathbf{S}}_0 = \mathbf{A}^\dagger \mathbf{X} (\mathbf{B}^\dagger)^T \quad (11)$$

Proof. This is equivalent to $\text{vec}(\hat{\mathbf{S}}_0) = \text{vec}\{\mathbf{A}^\dagger \mathbf{X} (\mathbf{B}^\dagger)^T\} = (\mathbf{B}^\dagger \otimes \mathbf{A}^\dagger)\text{vec}(\mathbf{X}) = (\mathbf{B} \otimes \mathbf{A})^\dagger \mathbf{x} = \Phi^\dagger \mathbf{x}$. \square

Using the above theorems and with trivial modifications of the other steps of SL0, the final 2-dimensional version of SL0 (called 2D-SL0) is obtained as Fig. 1.

4. EXPERIMENTAL RESULTS

In this section, we consider the performance of 2D-SL0 experimentally. In our experiment, we generate two random matrices $\mathbf{A} \in \mathbb{R}^{40 \times 100}$ and $\mathbf{B} \in \mathbb{R}^{50 \times 100}$, so with a probability equal to one we have $\text{spark}(\mathbf{A}) = 41$ and $\text{spark}(\mathbf{B}) = 51$. The sparse matrix $\mathbf{S} \in \mathbb{R}^{100 \times 100}$ is artificially created using the following model: k elements out of 10000 elements of \mathbf{S} are randomly chosen, *without imposing any restriction on their positions*, to be active, and the remaining inactive. The values of the active and inactive elements are drawn from a zero mean Gaussian random variable, with variances σ_{on}^2 and σ_{off}^2 , respectively. σ_{off} is to model the noise in the sources. In our simulation, we set $\sigma_{\text{on}} = 1$ and σ_{off} has two values .01 and .001. The matrix \mathbf{X} is then generated using $\mathbf{X} = \mathbf{A}\mathbf{S}\mathbf{B}^T$. To evaluate the estimation quality, Signal-to-Noise Ratio (SNR) is used which is defined as $10 \log(\|\mathbf{S}\|_F^2 / \|\mathbf{S} - \hat{\mathbf{S}}\|_F^2)$ where $\hat{\mathbf{S}}$ is the estimation of the matrix \mathbf{S} . Figure 2 shows the output

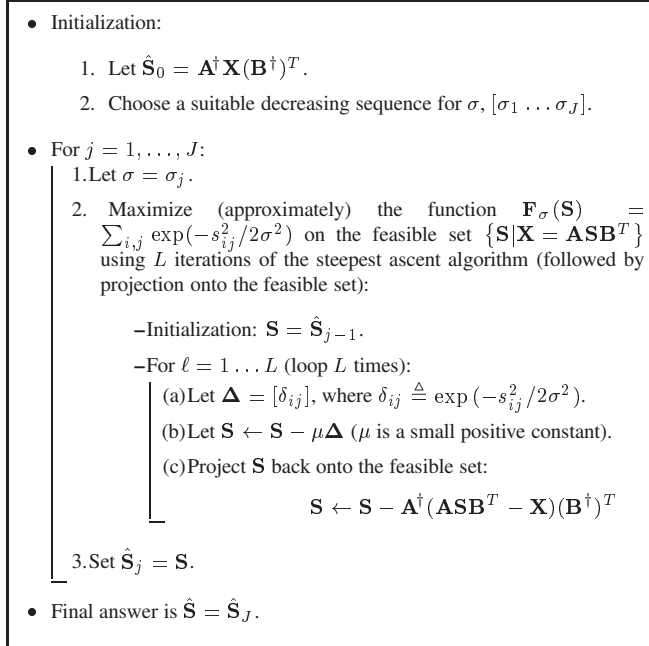


Fig. 1. Two-dimensional SL0 algorithm (2D-SL0).

SNR as a function of k , obtained by averaging SNR's over 100 runs of the algorithm, for different randomly generated \mathbf{A} , \mathbf{S} and \mathbf{B}

In a typical run of the algorithm, 2D-SL0 took 0.9 seconds, while 1D SL0 on the converted problem (3) took about 40 seconds on the same computer. Consequently, 2D-SL0 can reduce the complexity drastically.

It is interesting to note that as it is seen in Fig. 2, 2D-SL0 works well where k is smaller than a critical value which is approximately equal to $\text{spark}(\mathbf{A})\text{spark}(\mathbf{B})/4 \approx 500$, while we had shown at the end of Section 3 that $\text{spark}(\mathbf{A})\text{spark}(\mathbf{B})/4$ is only a necessary (not sufficient) condition for uniqueness. This critical value is much bigger than the maximum $\|\mathbf{S}\|_0$ given by $\Phi\mathbf{s} = \mathbf{x}$, that is, $\min(\text{spark}(\mathbf{A}), \text{spark}(\mathbf{B}))/2 = 20.5$.

5. CONCLUSION

In this paper, we considered the 2D sparse decomposition problem, and we discussed the uniqueness of the representation. Moreover, we obtained the 2D-SL0 algorithm by modifying SL0, to practically calculate this sparse representation. The computational load and required memory for this approach is highly less than the trivial approach of converting this 2D problem to a 1D problem.

Future works include extension of this model to higher dimensions, as well as testing the model in different applications such as image denoising and compression.

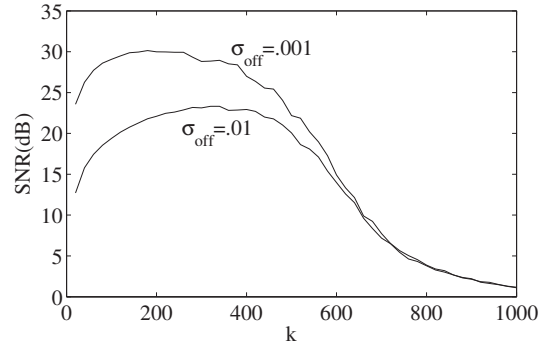


Fig. 2. Average SNRs (over 100 runs of the algorithm) as a function of k , the number of active elements of \mathbf{S} , for two values of σ_{off} .

6. REFERENCES

- [1] S.S. Chen, D.L. Donoho, and M.A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–59, January 2001.
- [2] S. Mallat and Z. Zhang, "Matching pursuits with time frequency dictionaries," *IEEE Trans. on Signal Proc.*, vol. 41, no. 12, pp. 3397–3415, 1993.
- [3] R. Gribonval and S. Lesage, "A survey of sparse component analysis for blind source separation: principles, perspectives, and new challenges," in *Proceedings of ESANN'06*, April 2006, pp. 323–330.
- [4] D. L. Donoho, "Compressed sensing," *IEEE Trans. on Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.
- [5] D. L. Donoho and M. Elad, "Maximal sparsity representation via ℓ^1 minimization," *the Proc. Nat. Aca. Sci.*, vol. 100, no. 5, pp. 2197–2202, March 2003.
- [6] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using focuss, a reweighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, March 1997.
- [7] R. Gribonval and M. Nielsen, "Sparse decompositions in unions of bases," *IEEE Trans. Inform. Theory*, vol. 49, no. 12, pp. 3320–3325, Dec. 2003.
- [8] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed ℓ^0 norm," *IEEE Trans. on Signal Proc.*, vol. 57, no. 1, pp. 289–301, January 2009.
- [9] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Prentice Hall, 2007.
- [10] K.B.Petersen and M.S.Pedersen, *The Matrix Cookbook*, 2008, url=<http://matrixcookbook.com>.