

A Fixed-Point Iterative Schema for Error Minimization in k -Sparse Decomposition

Alessandro Adamo

Dipartimento di Matematica
Università degli Studi di Milano
Via Saldini 50, I-20133 Milan, Italy
Email: alessandro.adamo@unimi.it

Giuliano Grossi

Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
Via Comelico 39, I-20135 Milan, Italy
Email: grossi@dsi.unimi.it

Abstract—Analogously to the well known greedy strategy called Orthogonal Matching Pursuit (OMP), we present a new algorithm to solve the sparse approximation problem over redundant dictionaries where the input signal is restricted to be a linear combination of k atoms or fewer from a fixed dictionary. The basic strategy of our method rests on a family of nonlinear mappings which results to be contractive in a interval close to zero. By iterating contractions and projections the method is able to extract the most significant components also for noisy signal which subsumes an ideal underlying signal having sufficiently sparse representation. For reasonable error level, the fixed point solution of such a iterative schema provides a sparse approximation containing only the nonzero terms characterizing the unique sparsest representation of the ideal noiseless sparse signal. The heuristic method so derived has been applied both to synthetic and real data. The former was generated by combining exact signals drawn by usual Bernoulli-Gaussian model and Gaussian noise; the later is taken by electrocardiogram (ECG) signals with application to the dictionary learning problem. In both cases the proposed method outperforms OMP method both regarding sparse approximation error and computation time.

I. INTRODUCTION

In recent years, a lot of research and application activities have been carried out examining sparsity in signal processing and adjoining areas [1], [2]. Unfortunately to obtain the best sparse approximations for large signal dimensions is non polynomial (NP-hard).

In the spirit of this challenge, here we show the design of a fast algorithm which calculates a nearly-optimal sparse representation of an arbitrary noisy input signal. The method rests on nonlinearities in order to promote sparsity followed by projections into linear spaces of fixed low dimensions with reduced reconstruction error. In particular, the method consists on a refinement of a more general technique [3] based on two kinds of Lipschitzian mappings: an asymptotically non-expansive [4] parametric family of nonlinear functions able to select near-feasible solutions having the sparsity property and a nonexpansive linear map consisting on orthogonal projection of near-feasible point along the space of the affine solutions of an inhomogeneous linear system. The main purpose of this interaction is that of reducing the gap existing between the two image spaces, being the first a subspace containing sparse points (in the sense of the ℓ^0 -norm which measures the total number of nonzero elements) while the second is

a set containing infinite solutions for the problem at hand. By composing the two mappings we face nonlinear problems in Banach space for which fixed-points exist. Moreover, they may be obtained as the limit of a fixed-point iteration scheme defined by repeated images under the mapping of an arbitrary starting point in the space.

As element of comparison we use the popular orthogonal matching pursuit algorithm (OMP) [5], [6], [7], [8], which has been widely used with the purpose of recovery sparse noisy signals. OMP is an iterative greedy algorithm that selects at each step the dictionary element best correlated with the residual part of the signal. Then it produces a new approximation by projecting the signal onto those elements which have already been selected. This technique just extends the trivial greedy algorithm which succeeds for an orthonormal system. Compared with other alternative methods, a major advantage of the OMP is its simplicity and fast implementation.

To put in evidence the performances of our method with respect to those exhibited by OMP we do two kinds of tests. The first considers randomly generated synthetic instances of a noisy linear system satisfying sparsity requirements. The second is an application of dictionary learning aimed to adapt dictionaries in order to achieve sparse signal representations. Known techniques in this regards are KSVD [9], which is a generalizing the K-means algorithm and the Method of Optimal Directions (MOD) [10]. The basic waveforms used both to learn dictionaries and to test signals outside the training sets are electrocardiogram (ECG) signals. We choose this application because, as required in many cases, it is of practical importance to fix a priori the desired level of sparsity in sparse recovery algorithms, especially when an analytic dictionary that fits a signal class is not known. To do this one usually relies on several examples of signals chosen in the class and sets a dictionary that minimizes the approximation error under a specified sparsity constraint. This constraint may be easily imposed by sparsity promotion methods that belong to the greedy class like Matching Pursuit [11], Orthogonal Matching Pursuit and Stagewise Orthogonal Matching Pursuit [12].

II. SPARSE REPRESENTATION

Let consider the problem of finding the sparsest representation possible in an overcomplete dictionary $\Phi = [\phi_1, \dots, \phi_m]$

assumed to be a collection of $m > n$ atoms or vectors in \mathbb{R}^n . A sparse representation for a given vector or signal $s = (s_1, \dots, s_n)$ expressed as a linear combination of atoms, i.e., $s = \sum_i \alpha_i \phi_i$, is measured in terms of the so-called ℓ^0 -norm $\|\alpha\|_0$, simply representing the number of non-zero elements in α . Usually, the ℓ^0 -norm is defined as the cardinality of set $\mathcal{S}(\alpha) = \{k : \alpha_k \neq 0\}$, called the support of α .

More generally, it is not sensible to assume that the available data s obey precise equality $s = \Phi\alpha$ with a sparse representation $\|\alpha\|_0 = k \ll n$. A more plausible scenario assumes sparse approximate representation in which there is an ideal noiseless signal s (admitting a sparse representation) corrupted by noise, leading to the following model:

$$s = \Phi\alpha + \varepsilon, \quad (1)$$

in which error or noise $\varepsilon \in \mathbb{R}^n$ gives rise, for instances, to measurements or estimates.

Adopting this noisy setting, the general goal of finding the sparsest decomposition of the signal s can be rephrased as the constrained minimization problem

$$\min_{\alpha \in \mathbb{R}^m} \|s - \Phi\alpha\|^2 \quad \text{subject to} \quad \|\alpha\|_0 \leq k, \quad (\text{P}_0)$$

where $\|\cdot\|$ denotes the ℓ^2 -norm.

However, this optimization problem is generally NP-hard. Therefore one seeks computationally efficient algorithms that can approximately solve (P₀), with the goal of recovering as sparse as possible coefficient vectors α . One of the most popular algorithms in this area which accomplish this task easily and quickly is the so called Orthogonal Matching Pursuit algorithm (OMP) together with some of its variants as Stagewise Orthogonal Matching Pursuit (STOMP).

The heuristic strategy adopted in OMP builds up k -element approximate representations a step at a time, adding to an existing $(k - 1)$ -element approximation a new term chosen in a greedy fashion to minimize the resulting ℓ^2 error (over all possible choices of the single additional term). When stopped after a suitable number of stages (fewer than m), one gets a sparse approximate representation. In practice, OMP updates the residuals by projecting the observation s onto the linear subspace spanned by the columns that have already been selected, and the algorithm then iterates. A more sophisticated approach and then much performing, which replaces the original sparse approximation problem by a linear programming problem is the well-known Basis Pursuit [2] technique.

III. SPARSITY PROMOTION BY LIMAPS ALGORITHM

This section is taken from [3] and is summarized here for completeness while the following section presents a variant of LIMAPS which is innovative for both the points of view, modeling and applications.

LIMAPS is a new sparse approximation technique which consists on a fixed-point iteration schema based on a nonlinear mapping aimed to uniformly enhance the sparseness level of each iterate. In fact, at each iteration step, every coefficient is

either contracted (moved towards zero) or preserved (moved far from zero) in order to project the new coefficient vector onto the linear subspace spanned by the columns of the matrix, which is the affine space associated to the linear model (1).

To deal with high dimensional data, the method provides a parametric family of nonlinear functions $\mathcal{F} = \{f_\lambda : \mathbb{R}^m \rightarrow \mathbb{R}^m \mid \lambda \in \mathbb{R}^+\}$ where a component is defined as:

$$f_\lambda(\alpha) = \alpha \odot \left(1 - e^{-\lambda|\alpha|}\right), \quad (2)$$

being \odot the Hadamard (elementwise) product. An explanation of the fact that each independent variable α_i is here multiplied by the weight or penalty mono-valued function $g_\lambda(|\alpha_i|) = 1 - e^{-\lambda|\alpha_i|}$ can be found upon close inspection of the asymptotic behavior of g_λ , i.e., when $\lambda \rightarrow +\infty$. It becomes evident that the penalty vanishing in the limit becoming the identity function, as showed by the

$$\lim_{\lambda \rightarrow +\infty} g_\lambda(|\alpha_i|) = \begin{cases} 0, & \text{if } \alpha_i = 0 \\ 1, & \text{when } \alpha_i \neq 0 \end{cases}.$$

As a consequence

$$\lim_{\lambda \rightarrow +\infty} |f_\lambda(\alpha_i)| = \alpha_i \quad \text{for all } \alpha_i \in \mathbb{R}.$$

Combining nonlinear mappings belongs to the family (2) and orthogonal projections in the null space of matrix Φ , it can be showed that the sequence $\{\alpha^{(t)}\}$ generated by the following fixed-point schema always converges:

$$\alpha^{(t+1)} = T_{\lambda_t}(\alpha^{(t)}) = P f_{\lambda_t}(\alpha^{(t)}) + \Phi^\dagger s, \quad (3)$$

where $P = I - Q$, with $Q = \Phi^\dagger \Phi$, is the orthogonal projector onto the null space of Φ , $\Phi^\dagger = (\Phi^T \Phi)^{-1} \Phi^T$ is the Moore-Penrose pseudo-inverse of Φ and $\{\lambda_t\}$ a suitable sequence satisfying $\sum_{t=0}^{\infty} 1/\lambda_t < +\infty$.

By letting $\lambda_t \rightarrow +\infty$, we derive the LIMAPS heuristic which is a finite fixed-point iterative procedure that acts as local minimizer of the separable variable function $g_\lambda(|\alpha|) = \sum_i 1 - e^{-\lambda|\alpha_i|}$ which is a continuous approximation of $\|\alpha\|_0$.

IV. THE k -LIMAPS ALGORITHM

The main drawback of LIMAPS is represented by the need of providing a right sequence for the parameter λ indexing the function family in (2) so as to achieve a convergent sequence $\{\alpha^{(t)}\}_{t \geq 0}$. Moreover, even if it has good performances on sparse recovery, in general there is no way to upper bound the sparsity level which LIMAPS carry out, up to accomplish an unnatural thresholding on the final coefficient vector.

In this section we suggest a new fixed point iterative method inspired by the same nonlinear operator which LIMAPS is based on. In particular, it is shown that the method is able to adaptively find a suitable sequence $\{\lambda_t\}$ for approximately solving the problem P₀. This choice should also be made in the light of relevant constraints imposed on the objective function of such a problem, that is to choose k coefficients not null and discard the remaining $m - k$.

A. Definition of Parameter λ

Fixed $1 \leq k \leq n$, a possible strategy for finding k -sparse solutions using LIMAPS consists on choosing $\lambda_t = \sigma_t^{-1}$ at time $t \geq 0$ satisfying

$$\sigma_t = \hat{\alpha}_{k+1}^{(t)}$$

being $\hat{\alpha}^{(t)}$ the absolute values of $\alpha^{(t)}$ rearranged in descending order and $\hat{\alpha}_{k+1}^{(t)}$ its k -th element. The goal of this choice is double:

- 1) to speed up the process aimed to drop the smallest coefficients, i.e., those corresponding to elements $\hat{\alpha}_j \leq \sigma_t$, which have indexes in the set $\Lambda^{(t)} = \{j : |\alpha_j^{(t)}| \leq \sigma_t\}$;
- 2) to minimize the solution error induced by $\alpha^{(t)}$ “adjusting” the not null coefficients, i.e., those corresponding to elements $\hat{\alpha}_j > \sigma_t$ which have indexes in the set $\Lambda_c^{(t)} = \{j : |\alpha_j^{(t)}| > \sigma_t\}$.

Based upon this strategy, the method should ideally force the σ_t values in such a way to have

$$\lim_{t \rightarrow +\infty} g_{\sigma_t}(|\alpha_j|) = 1 - e^{-|\alpha_j|/\sigma_t} = \begin{cases} 1, & \text{if } j \in \Lambda^{(t)} \\ 0, & \text{if } j \in \Lambda_c^{(t)}. \end{cases}$$

Clearly, this requires that for $\forall j \in \Lambda^{(t)}$ the ratio between the absolute value of the coefficient $|\alpha_j^{(t)}|$ and the parameter σ_t tends to infinite, while $\forall j \in \Lambda_c^{(t)}$, $|\alpha_j^{(t)}|$ must be infinitesimal of order greater than that of σ_t , leading to

$$\lim_{t \rightarrow +\infty} f_{\sigma_t}(\alpha_j^{(t)}) = \alpha_j^{(t)} g_{\sigma_t}(|\alpha_j^{(t)}|) = \begin{cases} \alpha_j^{(t)}, & \text{if } j \in \Lambda^{(t)} \\ 0, & \text{if } j \in \Lambda_c^{(t)}. \end{cases}$$

B. Convergence Issue

The LIMAPS algorithm computes fixed-points of iterated nonlinear functions for sparse recovery that are solutions of the exact (without noise) model $\Phi\alpha = s$. The noisy case described by (1) can be recast as exact model for the noisy signal $\bar{s} = s - \varepsilon$. Therefore, in the noisy setting the trajectory followed by the dynamical system belongs to the affine space $\{\alpha \in \mathbb{R}^m : \Phi\alpha = \bar{s}\}$ associated to linear system (1).

Starting by these considerations and making the above choice for parameter λ , if we rewrite (3) as

$$\alpha^{(t+1)} = \alpha^{(t)} - Q\alpha^{(t)} - P \left[\alpha^{(t)} \odot e^{-|\alpha^{(t)}|/\sigma_t} \right] + \Phi^\dagger \bar{s},$$

since at each step $t > 0$ holds $Q\alpha^{(t)} = \Phi^\dagger \bar{s}$, the present iterative system has dynamics described by motion equation

$$\alpha^{(t+1)} - \alpha^{(t)} = P \left[\alpha^{(t)} \odot e^{-|\alpha^{(t)}|/\sigma_t} \right]. \quad (4)$$

A typical issue that arises in the field of dynamical systems like that described by recurrence equation (4) is the analysis of convergence. The question here may be stated in terms of convergence in norm of the sequence $\{\alpha^{(t)}\}$ which amounts for instance the requirement

$$\lim_{t \rightarrow +\infty} \left\| \alpha^{(t+1)} - \alpha^{(t)} \right\| = 0$$

and thus in turn implies

$$\left\| P \left[\alpha^{(t)} \odot e^{-|\alpha^{(t)}|/\sigma_t} \right] \right\| \rightarrow 0 \quad \text{as } t \rightarrow +\infty. \quad (5)$$

To provide empirical evidence on the convergence ratio, in Fig. 1 we plot the curves given by the norm in (5) during the first simulation steps of system (4). They are chosen as examples for highlighting how it behaves and how is in general the slope of the curves which result to be decaying in all simulations. Here in particular, k -sparse random instances $s \in \mathbb{R}^n$ and random matrix dictionaries $\Phi \in \mathbb{R}^{n \times m}$ with fixed size $n = 100$ and various $m = 200, \dots, 1000$ have been used. The different slopes are mainly due to the ratio m/n rather than the values imposed to the algorithm by means of the sparsity parameter k . In fact, the curves do not significantly change when we use values for $k > k^*$, where k^* is the optimum sparsity of the given signal s .

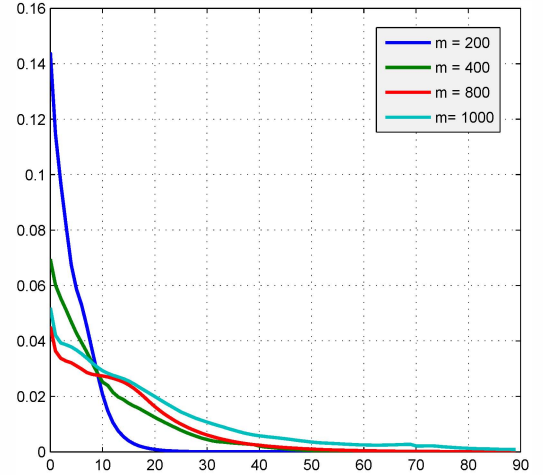


Fig. 1. Plotting of the norm in (5) with sparsity $k = 10$, size $n = 100$ and $m = 200, 400, 800, 1000$.

C. The algorithm

The overall algorithm, we call k -LIMAPS (which stands for k -COEFFICIENTS LIPSCHITZIAN MAPPINGS FOR SPARSITY), is sketched in Algorithm 1.

It should be noted that the last step of the algorithm accomplishes a thresholding of the final point carried out by while loop because in some cases it can have some noise among the null coefficients, that is those with indexes in the set Λ_c . However, experimentally we found that such a coefficients reach arbitrary close to zero values as the number of loops increases, making the threshold step not every useful. This annealing-like behavior which hits the not required coefficients exhibited by k -LIMAPS already at the beginning of the first iterations, is captured in the plotting of Fig. 2.

The figure shows a typical behavior of k -LIMAPS : as the time increases the values of σ_t decrease up to establish a clear separation between null and not null coefficients. Numerically,

Algorithm 1 k -LIMAPS

Require: - a dictionary $\Phi \in \mathbb{R}^{n \times m}$
 - its pseudo-inverse Φ^\dagger
 - a signal $s \in \mathbb{R}^n$
 - a sparsity level k

```

1:  $\alpha \leftarrow \Phi^\dagger s$ 
2: while [cond] do
3:    $\sigma \leftarrow \text{sort}(|\alpha|)$            <descending order coefficients>
4:    $\lambda \leftarrow 1/\sigma_k$        <sparsity ratio update>
5:    $\beta \leftarrow f_\lambda(\alpha)$      <increase sparsity>
6:    $\alpha \leftarrow \beta - \Phi^\dagger(\Phi\beta - s)$  <orthogonal projection>
7: end while
8:  $\alpha_j \leftarrow 0 \quad \forall j \text{ s.t. } |\alpha_j| \leq \sigma_k$  <thresholding>
  
```

Ensure: an approx. solution of $s = \Phi\alpha$ s.t. $\|\alpha\|_0 \leq k$

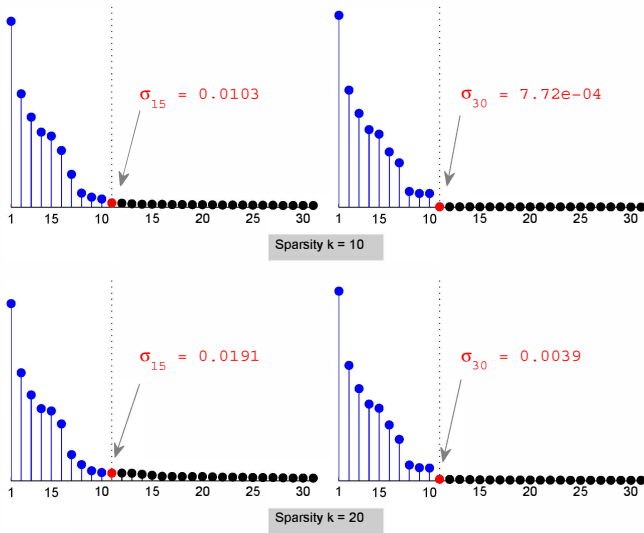


Fig. 2. Sorted absolute values of the α coefficients. The red stencils represent the absolute values of σ_t at various times. They separate the null coefficients (black stencils) from the absolute values of those not null (blue stencils).

the instance reported in the figure is related to a 10-sparse signal of length $n = 100$ on a dictionary of size $m = 800$ in which the input sparsity parameter k of the algorithm is fixed to 10 and 20 respectively. Thus the values of σ_t (here $t = 15$ and 30 are reported) become close to zero within few steps, with a delay in the latter case, i.e., when we use $k = 20$ as input for k -LIMAPS.

V. EMPIRICAL RESULTS

In order to empirically study how the proposed algorithm performs, we have carried out two kinds of experiments on synthetic and real data respectively. The first was conducted on random instances assumed to have the sparsity property, while the second was aimed to learn a dictionary for a class of electrocardiogram (ECG) signals taken from standard benchmark. Other applications that can benefit from the sparsity and overcompleteness concepts include compression, regularization in inverse problems and feature extraction.

Our experiments were performed on AMD Athlon II X4 630 Processor 64 bit, 2.8 GHz processor with 4 GB of memory, using MATLAB in conjunction with SparseLab, Sparse Optimization Toolboxes and KSVD-Toolbox for algorithms implementation. The algorithm k -LIMAPS is available online at the URL <http://dalab.dsi.unimi.it/klimaps>.

A. Comparison on Random Instances

By synthetic instances we mean a collection of instances of problem P_0 satisfying sparsity requirements and defined by an ensemble of matrices Φ of size $n \times m$ and an ensemble of k -sparse vectors $s \in \mathbb{R}^n$. All matrices have been sampled from the uniform spherical ensemble, while each vector s was a single realization of a random variable having k nonzeros sampled from a standard iid $\mathcal{N}(0, 1)$ distribution.

The performances of the two algorithms on each realization are measured according to the quantitative criterion given by the mean square error:

$$\text{MSE} = \frac{\|\Phi\alpha - s\|^2}{n}$$

A diagram of the integral of the error depicts the performances of the two algorithms for a wide variety of instances. The average value of such cumulative error measure is displayed as a function of $\rho = k/n$ and $\delta = n/m$. Fig. 3 displays a grid of $\delta - \rho$ values, with δ ranging through 50 equispaced points in the interval $[.01, .5]$ and ρ ranging through 100 equispaced points in $[.01, 1]$; here the signal length is fixed to $n = 100$. Each point on the grid shows the cumulated mean square error between the original and reconstructed, averaged over 100 independent realizations at a given k, m .

It can be noticed that MSE of OMP increases particularly when δ tends to .5 and ρ tends to 1, while k -LIMAPS is less sensitive with respect to these saturation values.

B. Comparison on ECG Dictionary Learning

To show the effectiveness of our algorithm on real data, we focus on the dictionary learning task for sparse representation applied to ECG signals. Instances are taken from the Physionet bank [13], specifically in the class of normal sinus rhythm, collecting many patient records with normal cardiac activity. We took a long ECG registration relative to a single patient and we split the signal into segments of length $n = 128$, each one corresponding to a second of the signal registration and sampled with frequency $f_s = n$, then we divide the blocks so obtained into two groups: training set and test set.

To perform the dictionary learning task we use KSVD and MOD techniques working in conjunction with both the pursuit algorithm OMP and our nonlinear method k -LIMAPS as sparsity recovery algorithms. In the training phase, the algorithms perform 50 iteration steps with a fixed sparsity level of 64 coefficients (50% of the signals length), over a dataset collecting 512 samples randomly picked from training set. At the end of the learning phase, the dictionaries carried out by the learning algorithms were tested on 5000 signals picked from the test set using the same sparse recovery algorithm (OMP or k -LIMAPS) previously applied in the training phase.

TABLE II
PRDN OVER 5000 TEST SIGNALS.

	PRDN mean (%)	PRDN std. dev.
KSVD-LiMapS	16.17	5.26
MOD-LiMapS	15.86	5.05
KSVD-OMP	17.92	5.13
MOD-OMP	17.42	4.92

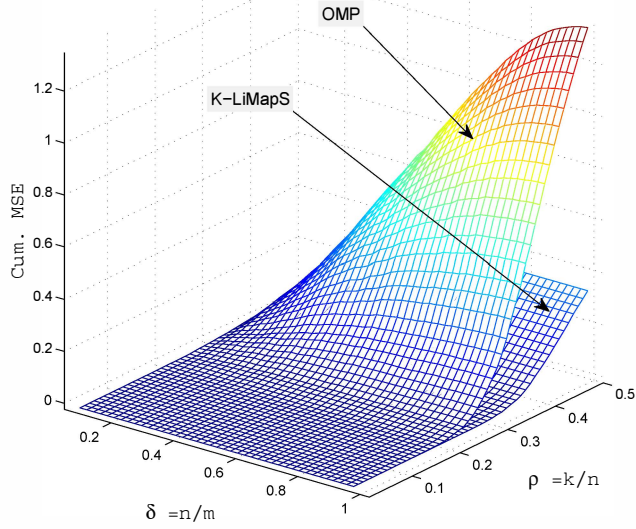


Fig. 3. Each point on the grid shows the cumulative MSE between the original s and reconstructed Φ_α signals, averaged over 100 independent realizations. The grid of $\delta - \rho$ values is done with δ ranging through 50 equispaced points in the interval $[.01, 5]$ and ρ ranging through 100 equispaced points in $[.01, 1]$.

To evaluate the accuracy of the signal reconstruction, one of the most used performance measure in the ECG signal processing field is the root mean square difference or PRD, together with its normalized version PRDN (which does not depend on the signal mean), defined respectively as:

$$\text{PRD} = 100 * \frac{\|s - \hat{s}\|_2}{\|s\|_2} \quad \text{and} \quad \text{PRDN} = 100 * \frac{\|s - \hat{s}\|_2}{\|s - \bar{s}\|_2},$$

where s and \hat{s} are the original and the reconstructed signals respectively, while \bar{s} is the original signal mean.

As it can be observed in Tables I and II our sparse recovery algorithm, applied to the dictionary learning, obtains the best results on average for both training algorithms MOD and KSVD, with standard deviations comparable to that of OMP.

TABLE I
PRDN OVER 5000 TEST SIGNALS.

	PRDN mean (%)	PRDN std. dev.
KSVD-LiMapS	15.86	5.26
MOD-LiMapS	16.16	5.05
KSVD-OMP	17.92	5.13
MOD-OMP	17.41	4.93

The convergence error is a parameter in evaluating such a kind of algorithms. In figure 4 are reported all MSEs ensured by the algorithms: also in this case k -LiMapS outperforms OMP with both MOD and KSVD algorithms.

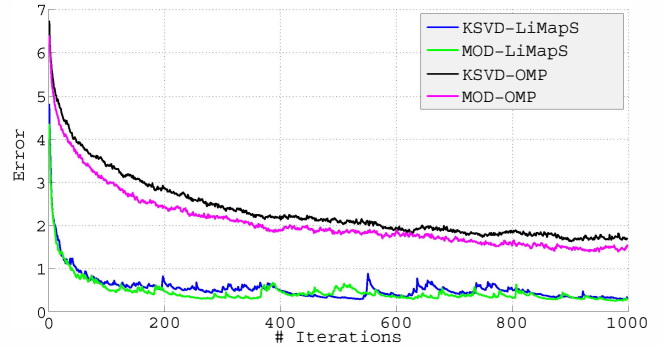


Fig. 4. Mean square error over the training set during each iteration of the learning process.

As an example, in Fig. 5 an ECG signal block and the corresponding recovered signal by means of the two algorithms KSVD-LiMapS and MOD-LiMapS are shown. Qualitatively speaking, the signals recovered using dictionaries trained with OMP suffer from a significant error in the more “flat” regions, which are mainly localized nearby the most prominent features of a normal electrocardiogram, given by the three graphical deflections seen on a typical ECG signal and called QRS complex.

VI. CONCLUSIONS

This work focuses on a sparsity heuristic for identifying the most significant components of a given noisy signal. In particular, we take in exam the case in which an ideal underlying signal admits a sparse representation over an overcomplete dictionary but we can observe only a noisy version. Even if, in general, the problem of finding sparse representations must be unstable in the presence of noise, we show that the method benefits of its ability to adaptively derive the values of the unique parameter that guarantees the stability required to extract the exact number of components fixed a priori.

The heuristic is derived by a version devised for ideal noiseless signal which consists on a fixed-point iteration scheme which alternates the application of a suitable nonlinear mapping to the points of the affine space associated to the undetermined system. The convergence as well as the stability of the fixed point found after a necessary transition phase is showed empirically observing how the ratio of the difference between successive iterates decreases. Theoretically, it can be shown that the sequence of points given by the trajectory of

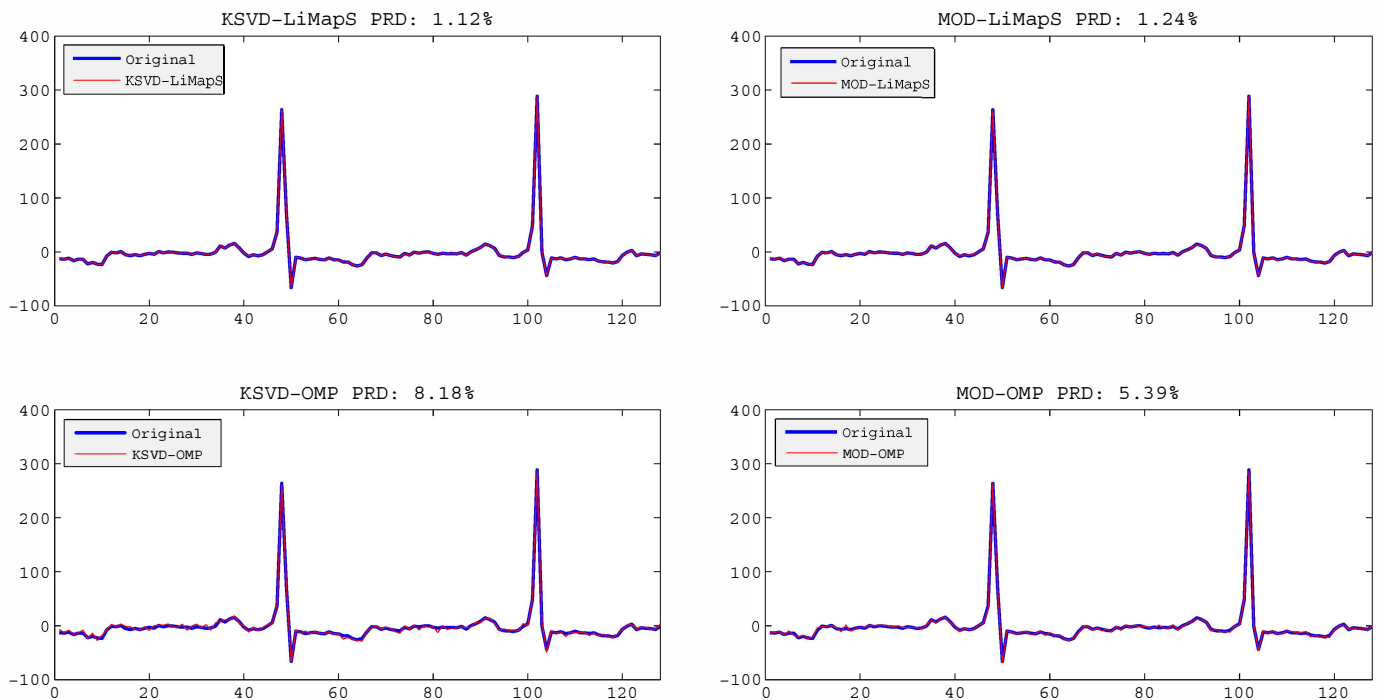


Fig. 5. Original and reconstructed signal from learned dictionaries. KSVD-LiMapS and MOD-LiMapS obtain less noise over the flat zone (nearly the QRS complex) if compared with the KSVD-OMP and MOD-OMP dictionary learning algorithms.

such a iterative process gives rise to a Cauchy sequence in an Hilbert space.

In the experimental section we showed the ability of our algorithm to approximately solve sparse recovery problem when the level of sparsity required is fixed in advance. We have considered both the case of random generated instances and the case of real data picked to ECG signal database with application to the dictionary learning. We directly compare all accomplished tests with the well-known greedy method called Orthogonal Matching Pursuit and we show that the proposed method outperforms the latter one obtaining less noisy solutions in both kinds of experiment.

REFERENCES

- [1] D. L. Donoho, For most large underdetermined systems of linear equations the minimal ℓ_1 -norm solution is also the sparsest solution, *Comm. Pure Appl. Math* 59 (2004) 797–829.
- [2] S. S. Chen, D. L. Donoho, M. A. Saunders, Atomic decomposition by basis pursuit, *SIAM Rev.* 43 (2001) 129–159.
- [3] A. Adamo, G. Grossi, Sparsity recovery by iterative orthogonal projections of nonlinear mappings, in: *Proceedings of the 11th IEEE International Symposium on Signal Processing and Information Technology (ISSPIT 2011)*, IEEE Computer Society, 2011, pp. 173–178.
- [4] K. Goebel, W. A. Kirk, A fixed point theorem for asymptotically nonexpansive mappings, *Proc. Amer. Math. Soc.* 35 (1972) 171–174.
- [5] Y. C. Pati, R. Rezaifar, Y. C. P. R. Rezaifar, P. S. Krishnaprasad, Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition, in: *Proceedings of the 27 th Annual Asilomar Conference on Signals, Systems, and Computers*, 1993, pp. 40–44.
- [6] J. A. Tropp, Anna, C. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, *IEEE Trans. Inform. Theory* 53 (2007) 4655–4666.
- [7] D. L. Donoho, M. Elad, V. N. Temlyakov, Stable recovery of sparse overcomplete representations in the presence of noise, *IEEE TRANS. INFORM. THEORY* 52 (1) (2006) 6–18.
- [8] J. A. Tropp, Greed is good: Algorithmic results for sparse approximation, *IEEE Trans. Inform. Theory* 50 (2004) 2231–2242.
- [9] M. Aharon, M. Elad, A. Bruckstein, K-svd : An algorithm for designing overcomplete dictionaries for sparse representation, *Structure* 54 (11) (2006) 4311–4322.
- [10] K. Engan, S. Aase, J. H. Husoy, Method of optimal directions for frame design, *Acoustics, Speech, and Signal Processing*, IEEE International Conference on 5 (1999) 2443–2446.
- [11] S. Mallat, Z. Zhang, Matching pursuit with time-frequency dictionaries, *IEEE Transactions on Signal Processing* 41 (1993) 3397–3415.
- [12] D. L. Donoho, Y. Tsaig, I. Drori, J. luc Starck, Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit, *Tech. rep.*, Stanford University (2006).
- [13] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, H. E. Stanley, Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals, *Circulation* 101 (23) (2000) e215–e220.