Fixed points of dictionary learning algorithms for sparse representations

Boris Mailhé, Mark D. Plumbley

Fixed points of dictionary learning algorithms for sparse representations

Boris Mailhé, *Member IEEE,SPS* and Mark D. Plumbley, *Senior Member, IEEE,SPS*

***Abstract*—This work provides theoretical arguments to compare dictionary learning algorithms for sparse representations. Three algorithms are considered: Sparsenet, MOD and K-SVD. The main theoretical result is that the fixed points of the Sparsenet and MOD dictionary update stages are the critical points of the residual error energy function (i.e. points with null gradient, not necessarily local minima), whereas the set of K-SVD fixed points is strictly included in the critical point set of the error energy. An example of a point is also provided where Sparsenet and MOD would stop whereas K-SVD can reach a solution with lower residual error. Further experiments show that the result of Sparsenet is a very good starting point for K-SVD. The combination of Sparsenet followed by K-SVD provides a significant improvement in terms of exact recovery rate and approximation quality.**

***Index Terms*—Machine learning algorithms, Dictionaries, Optimization, Sparse coding**

I. Introduction

In the method of sparse decompositions, a signal is represented by the linear combination of a few vectors named *atoms* chosen from a large set named a *dictionary*. Such representations are useful for many applications such as audio compression [1], image denoising [2], image and audio inpainting [3] or [4]. There is no finite universal dictionary that can represent every signal sparsely, so a dictionary has to be chosen according to the signals to be represented.

If a good candidate dictionary is not already known, then the dictionary can be learnt from training examples. Several dictionary learning algorithms have been proposed in the literature such as Sparsenet [5], the Method of Optimal Directions (MOD) [6] and K-SVD [7]. However it is not always clear how these algorithms behave and compare to each other. In the present paper we provide a theoretical comparison between the behaviors of these algorithms, and in particular we will investigate their fixed points. To gain insight on the asymptotic behavior at convergence we will consider a simplified problem where a given gives the support of the decomposition.

Our main result is that K-SVD has strictly fewer fixed points than Sparsenet and MOD. The global minima are fixed points for all three algorithms, so K-SVD therefore has strictly fewer suboptimal points (i.e. those that are not global minima) to converge to. It also follows that Sparsenet or MOD could be used to initialize K-SVD, and indeed we find the combination of Sparsenet followed by K-SVD outperforms any one of the algorithms used alone.

II. Problem and algorithms

Let $\mathbf{Y}$ be a $D \times N$ matrix of training data and $M < N$ a dictionary size. The dictionary learning problem is that of finding a dictionary $\mathbf{\Phi}$ of size $D \times M$ and sparse coefficients $\mathbf{X}$ of size $M \times N$ such that $\mathbf{Y} \approx \mathbf{\Phi X}$. In this work we use the following problem statement.

Problem 1 (Dictionary learning)**.** *Given a training matrix $\mathbf{Y}$, a dictionary size M and a sparsity level K, minimize the Frobenius energy cost function*

$$f(\mathbf{\Phi}, \mathbf{X}) = \|\mathbf{Y} - \mathbf{\Phi X}\|_{\mathrm{F}}^2 \tag{1}$$

under the constraints

$$\forall m \in [1, M],\ \|\boldsymbol{\varphi}_m\|_2 = 1 \tag{2}$$

$$\forall n \in [1, N],\ \|\mathbf{x}_n\|_0 \leq K \tag{3}$$

where $\boldsymbol{\varphi}_m$ is the m^{th} column of $\mathbf{\Phi}$ and $\|\mathbf{x}_n\|_0$ is the number of non-zero coefficients in $\mathbf{x}_n$, the n^{th} column of $\mathbf{X}$.

Sparsenet, MOD and K-SVD all minimize the cost function (1) by cycling through 2 steps. Those steps can roughly be described as:

1) a sparse approximation step to update $\mathbf{X}$,
2) a dictionary update step to update $\mathbf{\Phi}$.

Boris Mailhé and Mark Plumbley are with the Queen Mary University of London (QMUL), School of Electronic Engineering and Computer Science (EECS), Centre For Digital Music (C4DM), E1 4NS London, United Kingdom (e-mail: firstname.name@eecs.qmul.ac.uk).

This work was supported by the EPSRC Project EP/G007144/1 Machine Listening using Sparse Representations and by the EU FET-Open project FP7-ICT-225913-SMALL.

This work was presented in part at the Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS'11) in Edinburgh, Scotland.

The sparse approximation step 1 is common to all algorithms, while the dictionary update step 2 is different for each.

A. Sparse approximation

Sparse approximation has been widely studied and any existing algorithm can be used in that step, for example Matching Pursuit (MP) [8], Orthogonal Matching Pursuit (OMP) [9] or ℓ_1 penalization, also known as Basis Pursuit Denoising (BPDN) [10]. OMP in particular offers useful theoretical guarantees: after the decomposition the support always has a sparsity of K or less, the sub-dictionary selected to represent each signal has full rank and the residual is orthogonal to all selected atoms [9]. We therefore use OMP for the sparse approximaiton step in this work.

B. Dictionary update

The main difference between the algorithms studied here is the way that they update the dictionary. Both Sparsenet and MOD perform the dictionary update with fixed values of the coefficients, while K-SVD simultaneously updates an atom and the values of its non-zero sparse coefficients.

1) Sparsenet: Sparsenet [5] updates each atom of the dictionary successively using a projected fixed step gradient descent. The gradient of the cost function (1) with respect to the atom $\boldsymbol{\varphi}_m$ is $\mathbf{R}\mathbf{x}^{mT}$ with $\mathbf{R} = \mathbf{Y} - \mathbf{\Phi}\mathbf{X}$ the residual and $\mathbf{x}^m$ the m^{th} line of $\mathbf{X}$ that contains the coefficients of the atom $\boldsymbol{\varphi}_m$. The projection on the ℓ_2 sphere to satisfy the constraint (2) simply consists in renormalizing the atom. Overall, the update sequence for one atom is:

$$\mathbf{R} = \mathbf{Y} - \mathbf{\Phi}\mathbf{X} \tag{4}$$

$$\boldsymbol{\varphi}_m \leftarrow \boldsymbol{\varphi}_m + \alpha \mathbf{R}\mathbf{x}^{mT},\ \forall m \in [1, M] \tag{5}$$

$$\boldsymbol{\varphi}_m \leftarrow \frac{\boldsymbol{\varphi}_m}{\|\boldsymbol{\varphi}_m\|_2},\ \forall m \in [1, M] \tag{6}$$

where α is the fixed step size.

2) MOD: MOD [6] updates the whole dictionary in one step with the solution of the unconstrained least-square problem followed by the same renormalization of each atom as in Equation (6).

$$\mathbf{\Phi} \leftarrow \mathbf{Y}\mathbf{X}^+ \tag{7}$$

$$\boldsymbol{\varphi}_m \leftarrow \frac{\boldsymbol{\varphi}_m}{\|\boldsymbol{\varphi}_m\|_2},\ \forall m \in [1, M] \tag{8}$$

where $\mathbf{X}^+$ is the pseudo-inverse of $\mathbf{X}$.

3) K-SVD: K-SVD update each atom iteratively same as Sparsenet, but when updating an atom, it also updates the value of its corresponding non-zero coefficients [7]. The atom update problem then becomes a principal component problem. For the m^{th} atom $\boldsymbol{\varphi}_m$ and the corresponding coefficient line $\mathbf{x}^m$, the estimated contribution matrix $\mathbf{E}_{(m)}$ is defined as

$$\mathbf{E}_{(m)} = \left[\mathbf{R} + \boldsymbol{\varphi}_m \mathbf{x}^m\right]_{\mathrm{supp}(\mathbf{x}^m)},\ \forall m \in [1, M]. \tag{9}$$

Then the atom $\boldsymbol{\varphi}_m$ is updated with the principal component of its estimated contribution, and its non-zero coefficients $\mathbf{x}^m_{\mathrm{supp}(\mathbf{x}^m)}$ with the correlation of the new atom with the contribution:

$$\boldsymbol{\varphi}_m \leftarrow \underset{\mathbf{v}, \|\mathbf{v}\|_2 = 1}{\mathrm{argmax}}\ \mathbf{v}^T \mathbf{E}_{(m)} \mathbf{E}_{(m)}^T \mathbf{v},\ \forall m \in [1, M] \tag{10}$$

$$\mathbf{x}^m_{\mathrm{supp}(\mathbf{x}^m)} \leftarrow \boldsymbol{\varphi}_m^T \mathbf{E}_{(m)},\ \forall m \in [1, M]. \tag{11}$$

III. Learning with a known support

Each of the algorithms considered use an iteration split into two steps, but the location of the split differs between K-SVD and the other two algorithms. To fit all the algorithms in a common framework we need to consider that each algorithms has three tasks to perform, not only two:

1) estimate the support of $\mathbf{X}$,
2) estimate the non-zero values of $\mathbf{X}$,
3) estimate the dictionary $\mathbf{\Phi}$.

Sparsenet and MOD perform the first and second tasks together, then the third, while K-SVD performs the first task, then the second and third together. With everything else fixed, the steps 2 and 3 are both simple quadratic problems. On the other hand, the sparse representation problem (i.e. the sequence of steps 1 and 2) is known to be NP-Hard [11]. Therefore the step 1 itself is NP-Hard because if one can solve it in polynomial time,then one can solve the whole NP-Hard sparse representation problem in polynomial time by finding the best support first, then the best coefficients for this support. One can then speculate whether dictionary learning would become a simple problem if the support were known.

Forcing an algorithm to keep the same support is also a way to investigate its asymptotic behavior in the general case: since the support is a discrete variable, if it converges, then it reaches its final value after a finite number of iterations then does not change anymore.

A. Problem formulation

When learning with a given coefficient support $\mathbf{\Gamma} \in \{0, 1\}^{M \times N}$ such that $\gamma^m_n = 0 \Rightarrow x^m_n = 0$, the cost function to minimize remains the function f defined in Equation (1) The sparsity constraint (3) is replaced by a support constraint on $\mathbf{X}$. The problem can now be written as:

Problem 2 (Dictionary learning with a known support)**.** *Given a training matrix* $\mathbf{Y}$ *and a support matrix* $\mathbf{\Gamma}$*, minimize the cost function*

$$f(\mathbf{\Phi}, \mathbf{X}) = \min \|\mathbf{Y} - \mathbf{\Phi}\mathbf{X}\|_{\mathrm{F}}^2 \tag{12}$$

under the constraints

$$\forall m \in [1, M],\ \|\boldsymbol{\varphi}_m\|_2 = 1 \tag{13}$$

$$\mathrm{supp}(\mathbf{X}) \subseteq \mathrm{supp}(\mathbf{\Gamma})\ . \tag{14}$$

The $\mathbf{\Gamma}$ matrix can also be used to express the support of the n^{th} signal and the co-support of the m^{th} atom $\mathrm{supp}(\mathbf{x}^m)$ as respectively $\boldsymbol{\gamma}_n$ and $\boldsymbol{\gamma}^m$.

The support constraint (14) reduces the dimension of the problem: the scalar unknowns of Problem 2 are each entry φ_m^d of $\mathbf{\Phi}$ and the non-zero entries of $\mathbf{X}$ only, i.e. each entry x_n^m of $\mathbf{X}$ with m and n such that $\gamma_n^m = 1$. One could rewrite the cost function f using those variables only:

$$f(\mathbf{\Phi}, \mathbf{X}) = \sum_{n=1}^{N} \|\mathbf{y}_n - \mathbf{\Phi}_{\boldsymbol{\gamma}_n} \mathbf{x}_n^{\boldsymbol{\gamma}_n}\|_2^2\ . \tag{15}$$

Then the constraint (14) would not be needed anymore since the zero coefficients would not appear in the problem. However the notation (15) is less convenient since the matrix product $\mathbf{\Phi}\mathbf{X}$ in (12) has been broken into pieces. So we keep the current notation (12) with the whole $\mathbf{X}$ matrix but notice that its zero coefficients only serve as placeholders to preserve the shape of the matrix.

Problem 2 is easier than the original Problem 1 in Section II since it does not contain the non-convex sparsity constraint (3). However the cost function f remains a non-convex fourth degree multivariate polynomial because both $\mathbf{\Phi}$ and $\mathbf{X}$ are unknown.

B. Algorithms

The studied algorithms can be adapted to Problem 2 by replacing the sparse decomposition by a simple least-square optimization of the non-zero coefficients:

$$\mathbf{x}_n^{\boldsymbol{\gamma}_n} \leftarrow \mathbf{\Phi}_{\boldsymbol{\gamma}_n}^{+} \mathbf{y}_n \tag{16}$$

where $\mathbf{x}_n^{\boldsymbol{\gamma}_n}$ is the column containing the non-zero coefficients of the decomposition of the signal $\mathbf{y}_n$ over the selected sub-dictionary $\mathbf{\Phi}_{\boldsymbol{\gamma}_n}$. In this work we will refer to the known support versions of the studied algorithms as $\overline{\text{Sparsenet}}$, $\overline{\text{MOD}}$ and $\overline{\text{K-SVD}}$. Those algorithms are fully described in Algorithms 1, 2 and 3.

The projection (16) ensures that the decomposition of each signal over its sub-dictionary is orthogonal: for each signal $\mathbf{y}_n$ and each atom $\boldsymbol{\varphi}_m$, if $\mathbf{x}_n^m \neq 0$, then $\langle \mathbf{r}_n, \boldsymbol{\varphi}_m \rangle = 0$. As a consequence the residual $\mathbf{R}$ is orthogonal to the contributions of each atom for the Frobenius inner product $\langle \mathbf{A}, \mathbf{B} \rangle = \mathrm{Tr}(\mathbf{A}^T \mathbf{B})$:

$$\forall m \in [1, M], \langle \mathbf{R}, \boldsymbol{\varphi}_m \mathbf{x}^m \rangle = 0\ . \tag{17}$$

Algorithm 1 $(\mathbf{\Phi}, \mathbf{X}) = \overline{\text{Sparsenet}}\ (\mathbf{Y}, \mathbf{\Gamma})$, Sparsenet with a known support

```
while not converged do
    for all $n \in [1, N]$ do
        $\mathbf{x}_n^{\boldsymbol{\gamma}_n} \leftarrow \mathbf{\Phi}_{\boldsymbol{\gamma}_n}^{+} \mathbf{y}_n$
    end for
    for all $m \in [1, M]$ do
        $\mathbf{R} \leftarrow \mathbf{Y} - \mathbf{\Phi}\mathbf{X}$
        $\boldsymbol{\varphi}_m \leftarrow \boldsymbol{\varphi}_m + \alpha \mathbf{R} \mathbf{x}^{mT}$
        $\boldsymbol{\varphi}_m \leftarrow \frac{\boldsymbol{\varphi}_m}{\|\boldsymbol{\varphi}_m\|_2}$
    end for
end while
```

Algorithm 2 $(\mathbf{\Phi}, \mathbf{X}) = \overline{\text{MOD}}\ (\mathbf{Y}, \mathbf{\Gamma})$, MOD with a known support

```
while not converged do
    for all $n \in [1, N]$ do
        $\mathbf{x}_n^{\boldsymbol{\gamma}_n} \leftarrow \mathbf{\Phi}_{\boldsymbol{\gamma}_n}^{+} \mathbf{y}_n$
    end for
    $\mathbf{\Phi} \leftarrow \mathbf{Y}\mathbf{X}^{+}$
    for all $m \in [1, M]$ do
        $\boldsymbol{\varphi}_m \leftarrow \frac{\boldsymbol{\varphi}_m}{\|\boldsymbol{\varphi}_m\|_2}$
    end for
end while
```

IV. Fixed points of $\overline{\text{MOD}}$ and $\overline{\text{Sparsenet}}$

In this section we study the fixed points of $\overline{\text{Sparsenet}}$ and $\overline{\text{MOD}}$ under the known support constraint, i.e. the points that remain the same after one iteration of the Algorithms 1 and 2. We show that the fixed points of $\overline{\text{Sparsenet}}$ and $\overline{\text{MOD}}$ are the same as the critical points of the cost function f.

A. Critical points of the cost function

The critical points of a smooth real valued function are the points where its gradient is zero. The gradient is

Algorithm 3 $(\mathbf{\Phi}, \mathbf{X}) = \overline{\text{K-SVD}}\,(\mathbf{Y}, \mathbf{\Gamma})$, K-SVD with a known support

```
while not converged do
    for all $n \in [1, N]$ do
        $\mathbf{x}_n^{\gamma_n} \leftarrow \mathbf{\Phi}_{\gamma_n}^{+} \mathbf{y}_n$
    end for
    for all $m \in [1, M]$ do
        $\mathbf{R} \leftarrow \mathbf{Y} - \mathbf{\Phi}\mathbf{X}$
        $\mathbf{E}_{(m)} \leftarrow [\mathbf{R} + \boldsymbol{\varphi}_m \mathbf{x}^m]_{\gamma^m}$
        $\boldsymbol{\varphi}_m \leftarrow \underset{\mathbf{v}, \|\mathbf{v}\|_2 = 1}{\operatorname{argmax}}\ \mathbf{v}^T \mathbf{E}_{(m)} \mathbf{E}_{(m)}^T \mathbf{v}$
        $\mathbf{x}_{\gamma^m}^m \leftarrow \boldsymbol{\varphi}_m^T \mathbf{E}_{(m)}$
    end for
end while
```

obtained by differentiation of (12) with respect to both $\mathbf{\Phi}$ and $\mathbf{X}$:

$$\begin{aligned} df(\mathbf{\Phi}, \mathbf{X}) =& f(\mathbf{\Phi} + d\mathbf{\Phi}, \mathbf{X} + d\mathbf{X}) - f(\mathbf{\Phi}, \mathbf{X}) \\ =& \|\mathbf{Y} - (\mathbf{\Phi} + d\mathbf{\Phi})(\mathbf{X} + d\mathbf{X})\|_{\mathrm{F}}^2 - \|\mathbf{Y} - \mathbf{\Phi}\mathbf{X}\|_{\mathrm{F}}^2 \\ =& -2\langle \mathbf{R}\mathbf{X}^T, d\mathbf{\Phi} \rangle - 2\langle \mathbf{\Phi}^T \mathbf{R}, d\mathbf{X} \rangle \\ & + O(\|d\mathbf{X}\|_{\mathrm{F}}^2 + \|d\mathbf{\Phi}\|_{\mathrm{F}}^2)\ . \end{aligned}$$

where $\mathbf{R} = \mathbf{Y} - \mathbf{\Phi}\mathbf{X}$.

With a slight abuse of the partial derivative notation, we write the gradient components relative to $\mathbf{\Phi}$ and $\mathbf{X}$ in matrix form as

$$\frac{\partial f}{\partial \mathbf{\Phi}} = -2\mathbf{R}\mathbf{X}^T \tag{18}$$

$$\frac{\partial f}{\partial \mathbf{X}} = -2\mathbf{\Phi}^T \mathbf{R} \tag{19}$$

The critical points of f are the points where both of these components are zero. Moreover, Equation (19) is written with a full $\mathbf{X}$ matrix for ease of notation and to preserve the shape of the problem, but not all its coefficients are variables. So for Problem 2, the $\mathbf{X}$ gradient only has to be considered for the coefficients allowed to change by the constraint (14). So the critical points of f are defined by the conditions

$$\mathbf{R}\mathbf{X}^T = 0 \tag{20}$$

$$[\mathbf{\Phi}^T \mathbf{R}]_{\mathrm{supp}(\mathbf{\Gamma})} = 0\ . \tag{21}$$

B. Fixed point condition for the least-square coefficient update

Under the known support constraint, the update rule for the amplitude is given by Equation (16). The least-square coeffcient update is the first step of each iteration of all algorithms 1, 2 and 3. The condition for the coefficient matrix $\mathbf{X}$ to remain the same before and after that step is expressed as

$$\forall n \in [1, N], \mathbf{x}_n^{\gamma_n} = \mathbf{\Phi}_{\gamma_n}^{+} \mathbf{y}_n\ . \tag{22}$$

Lemma 1. *If all the sub-dictionaries $\mathbf{\Phi}_{\gamma_n}$ have full rank, then the critical point condition* (21) *is equivalent to the fixed point condition* (22)*.*

Proof: Let us consider the signal $\mathbf{y}_n$. If the sub-dictionary $\mathbf{\Phi}_{\gamma_n}$ has full rank K, then its pseudo-inverse $\mathbf{\Phi}_{\gamma_n}^{+}$ can expressed as $\mathbf{\Phi}_{\gamma_n}^{+} = \left(\mathbf{\Phi}_{\gamma_n}^T \mathbf{\Phi}_{\gamma_n}\right)^{-1} \mathbf{\Phi}_{\gamma_n}^T$. The fixed point condition (22) can then be expanded:

$$\mathbf{x}_n^{\gamma_n} = \mathbf{\Phi}_{\gamma_n}^{+} \mathbf{y}_n \tag{23}$$

$$\mathbf{x}_n^{\gamma_n} = \left(\mathbf{\Phi}_{\gamma_n}^T \mathbf{\Phi}_{\gamma_n}\right)^{-1} \mathbf{\Phi}_{\gamma_n}^T \mathbf{y}_n \tag{24}$$

$$\mathbf{\Phi}_{\gamma_n}^T \mathbf{\Phi}_{\gamma_n} \mathbf{x}_n^{\gamma_n} = \mathbf{\Phi}_{\gamma_n}^T \mathbf{y}_n \tag{25}$$

$$\mathbf{\Phi}_{\gamma_n}^T \left(\mathbf{y}_n - \mathbf{\Phi}_{\gamma_n} \mathbf{x}_n^{\gamma_n}\right) = 0 \tag{26}$$

$$\mathbf{\Phi}_{\gamma_n}^T \mathbf{r}_n = 0 \tag{27}$$

which is the zero gradient condition (21) for the n^{th} column $\mathbf{x}_n$ of $\mathbf{X}$. The previous equations can also be read from bottom to top to complete the equivalence proof. ■

The full-rank hypothesis is not restrictive in the complete learning case: if $\mathbf{\Phi}_{\gamma_n}$ does not have full rank, then there is a support strictly included in γ_n that gives an approximation of $\mathbf{y}_n$ with the same error as γ_n. Therefore γ_n is not a good support and the sparse approximation step should not select it. For example, OMP is proven to always select sub-dictionaries with full rank.

C. Fixed points for $\overline{\text{Sparsenet}}$

The dictionary update for $\overline{\text{Sparsenet}}$ is a fixed step projected gradient descent. An atom $\boldsymbol{\varphi}_m$ is fixed for one Sparsenet iteration if it remains the same after one gradient descent (5) followed by one renormalization (6), i.e. if the gradient descent preserves the direction of the atom. This can be written:

$$\exists \lambda_m > 0, \boldsymbol{\varphi}_m + \alpha \mathbf{R} \mathbf{x}^{mT} = \lambda_m \boldsymbol{\varphi}_m\ . \tag{28}$$

We can now investigate the link between the fixed point condition (28) and the critical point condition (20).

Theorem 1. *A point $(\mathbf{\Phi}, \mathbf{X})$ is a fixed point of $\overline{\text{Sparsenet}}$ if and only if it is a critical point of the cost function and it satisfies the normalization constraint* (13)*.*

Proof: The amplitude coefficients $\mathbf{X}$ are only updated during the least-square coefficient update, so the Lemma 1 can be applied directly and we only need to study the link between the critical point condition (20) and the fixed point condition (28).

The $\Leftarrow$ way of the proof is easy: if a point is critical for f, then the gradient for each atom is 0 so the atom remains the same after the gradient descent. If the atom is also normalized, then it is invariant for the whole Sparsenet iteration.

For the $\Rightarrow$ way, we need to check that the gradient descent and renormalization cannot cancel each other. Assume that the Criterion (28) holds for every atom $\boldsymbol{\varphi}_m$. By multiplying by $\boldsymbol{\varphi}_m^T$ and taking the trace, one gets:

$$\|\boldsymbol{\varphi}_m\|_2^2 + \alpha\langle \mathbf{R}, \boldsymbol{\varphi}_m \mathbf{x}^m\rangle = \lambda_m \|\boldsymbol{\varphi}_m\|_2^2 \tag{29}$$
$$\lambda_m = 1 \tag{30}$$

because $\|\boldsymbol{\varphi}_m\|_2 = 1$ and $\langle \mathbf{R}, \boldsymbol{\varphi}_m \mathbf{x}^m\rangle = 0$. One can then replace λ_m by 1 in Equation (28) and obtain

$$\boldsymbol{\varphi}_m + \alpha \mathbf{R}\mathbf{x}^{mT} = \boldsymbol{\varphi}_m, \ \forall m \in [1, M] \tag{31}$$
$$\mathbf{R}\mathbf{x}^{mT} = 0, \ \forall m \in [1, M] \tag{32}$$
$$\mathbf{R}\mathbf{X}^T = 0 \ . \tag{33}$$

■

That proof can also be interpreted geometrically: in general, for a smooth constrained optimisation problem, a fixed point of a projected gradient descent exists on the boundary of the admissible domain when the gradient of the cost function and of the constraint are collinear. In our case, the gradient of the ℓ_2 norm constraint is the atom itself and the gradient of the cost function f is always orthogonal to the atom, so they can only align when the gradient of f is 0.

D. Fixed points for the MOD dictionary update

For MOD, the update is performed with a pseudo-inverse (7) followed by a renormalization (8). A point is fixed for the MOD dictionary update if all the atoms are unchanged by the update, which can be expressed as:

$$\mathbf{Y}\mathbf{X}^+ = \boldsymbol{\Phi}\boldsymbol{\Lambda} \tag{34}$$

with $\boldsymbol{\Lambda}$ a positive definite diagonal matrix. If the coefficient matrix $\mathbf{X}$ has full rank M, then an argument similar to the one used in the proof of Theorem 1 can be used, as follows.

Theorem 2. *If the coefficient matrix* $\mathbf{X}$ *has full rank, then a point is fixed for MOD if and only if it is a critical point of the cost function* f *that satisfies the normalization constraint* (13).

Proof: As in Theorem 1, we only need to prove the link between the critical point condition (20) and the fixed point condition (34) because the coefficient update equivalence was already proven in Lemma 1.

If $\mathbf{X}$ has full rank, then its pseudo-inverse can be expressed as $\mathbf{X}^+ = \mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T\right)^{-1}$. One can easily prove that if the critical point condition (20) holds, then the dictionary is invariant for the pseudo-inverse update:

$$\mathbf{R}\mathbf{X}^T = 0 \tag{35}$$
$$\left(\mathbf{Y} - \boldsymbol{\Phi}\mathbf{X}\right)\mathbf{X}^T = 0 \tag{36}$$
$$\boldsymbol{\Phi}\mathbf{X}\mathbf{X}^T = \mathbf{Y}\mathbf{X}^T \tag{37}$$
$$\boldsymbol{\Phi} = \mathbf{Y}\mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T\right)^{-1} \tag{38}$$
$$\boldsymbol{\Phi} = \mathbf{Y}\mathbf{X}^+ \ . \tag{39}$$

So if a point $(\boldsymbol{\Phi}, \mathbf{X})$ is both critical and normalized, then its dictionary is invariant for the MOD update and renormalization. Conversely, we need to prove that the fixed point condition (34) implies the critical point condition (20). The proof follows the same structure as for Theorem 1: first prove that the fixed point condition (34) implies $\boldsymbol{\Lambda} = \mathbf{Id}$, then the result will follow. We start from the fixed point condition.

$$\mathbf{Y}\mathbf{X}^+ = \boldsymbol{\Phi}\boldsymbol{\Lambda} \tag{40}$$
$$\left(\boldsymbol{\Phi}\mathbf{X} + \mathbf{R}\right)\mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T\right)^{-1} = \boldsymbol{\Phi}\boldsymbol{\Lambda} \tag{41}$$
$$\boldsymbol{\Phi} + \mathbf{R}\mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T\right)^{-1} = \boldsymbol{\Phi}\boldsymbol{\Lambda} \tag{42}$$
$$\boldsymbol{\Phi}\mathbf{X}\mathbf{X}^T + \mathbf{R}\mathbf{X}^T = \boldsymbol{\Phi}\boldsymbol{\Lambda}\mathbf{X}\mathbf{X}^T \ . \tag{43}$$

We can now consider each column of Equation (43) separately. For all $m \in [1, M]$, we have:

$$\boldsymbol{\Phi}\mathbf{X}\mathbf{x}^{mT} + \mathbf{R}\mathbf{x}^{mT} = \boldsymbol{\Phi}\boldsymbol{\Lambda}\mathbf{X}\mathbf{x}^{mT} \ . \tag{44}$$

If we multiply by $\boldsymbol{\varphi}_m^T$ to the right and take the trace, Frobenius inner products appear:

$$\langle \boldsymbol{\Phi}\mathbf{X}, \boldsymbol{\varphi}_m\mathbf{x}^m\rangle + \langle \mathbf{R}, \boldsymbol{\varphi}_m\mathbf{x}^m\rangle = \langle \boldsymbol{\Phi}\boldsymbol{\Lambda}\mathbf{X}, \boldsymbol{\varphi}_m\mathbf{x}^m\rangle \tag{45}$$
$$\langle \boldsymbol{\Phi}\mathbf{X}, \boldsymbol{\varphi}_m\mathbf{x}^m\rangle = \langle \boldsymbol{\Phi}\boldsymbol{\Lambda}\mathbf{X}, \boldsymbol{\varphi}_m\mathbf{x}^m\rangle \tag{46}$$

because $\langle \mathbf{R}, \boldsymbol{\varphi}_m\mathbf{x}^m\rangle = 0$ for any m. $\boldsymbol{\Lambda} = \mathbf{Id}$ is obviously a solution to the system of M equations (46). To show that $\boldsymbol{\Lambda} = \mathbf{Id}$ the only solution, we need to reshape the system. For all $m \in [1, M]$, let $\mathbf{v}_m = \boldsymbol{\varphi}_m\mathbf{x}^m$. We then have

$$\boldsymbol{\Phi}\mathbf{X} = \sum_{m=1}^{M} \mathbf{v}_m \tag{47}$$
$$\boldsymbol{\Phi}\boldsymbol{\Lambda}\mathbf{X} = \sum_{m=1}^{M} \lambda_m \mathbf{v}_m \tag{48}$$

where the λ_m are the diagonal elements of $\boldsymbol{\Lambda}$. Let $\boldsymbol{\lambda} = \mathrm{diag}(\boldsymbol{\Lambda})$ be the column vector of length M containing those diagonal elements. Let $\mathbf{G} = \left(\langle \mathbf{v}_i, \mathbf{v}_j\rangle\right)_{1\leq i,j\leq M}$ be the Grammian of the $(\mathbf{v}_m)_{1\leq m\leq M}$ family. Then the system of M equations (46) can be rewritten as one matrix equation

$$\mathbf{G}\boldsymbol{\lambda} = \left(\langle \boldsymbol{\Phi}\mathbf{X}, \mathbf{v}_m \rangle\right)_{1 \leq m \leq M} \ . \tag{49}$$

Since $\mathbf{G}$ is a Gram matrix, its rank is the same as the rank of the $(\mathbf{v}_m)_{1 \leq m \leq M}$ indexed family. To prove that the $(\mathbf{v}_m)$ family has full rank M, we solve the equation of m unknown scalars δ_m

$$\sum_{m=1}^{M} \delta_m \mathbf{v}_m = 0 \tag{50}$$

which can also be written $\boldsymbol{\Phi}\boldsymbol{\Delta}\mathbf{X} = 0$ with $\boldsymbol{\Delta}$ an unkown diagonal matrix. The $\mathbf{X}$ matrix has full row rank M so

$$\boldsymbol{\Phi}\boldsymbol{\Delta}\mathbf{X} = 0 \Rightarrow \boldsymbol{\Phi}\boldsymbol{\Delta} = 0 \ . \tag{51}$$

The dicitonary $\boldsymbol{\Phi}$ has normalized columns so the norm of each column of $\boldsymbol{\Phi}\boldsymbol{\Delta}$ is $|\delta_m|$ and $\boldsymbol{\Phi}\boldsymbol{\Delta} = 0 \Rightarrow \boldsymbol{\Delta} = 0$. The $(\mathbf{v}_m)$ family and its grammian $\mathbf{G}$ both have full rank M. That proves that if the fixed point condition (34) holds, then $\boldsymbol{\Lambda} = \mathbf{Id}$. That value can now be injected into Equation (43):

$$\begin{aligned} \boldsymbol{\Phi}\mathbf{X}\mathbf{X}^T + \mathbf{R}\mathbf{X}^T &= \boldsymbol{\Phi}\mathbf{X}\mathbf{X}^T && (52) \\ \mathbf{R}\mathbf{X}^T &= 0 \ . && (53) \end{aligned}$$

■

E. The rank deficient case

If $\mathbf{X}$ does not have full rank, then the equivalence in Theorem 2 no longer holds. However the following lemma shows that in that case, unless the residual error is already 0, the point $(\boldsymbol{\Phi}, \mathbf{X})$ cannot be a global minimum because one can construct a strictly better point.

Lemma 2. *For any point $(\boldsymbol{\Phi}, \mathbf{X})$ where the coefficient matrix $\mathbf{X}$ does not have full rank, there exists another point $(\tilde{\boldsymbol{\Phi}}, \tilde{\mathbf{X}})$ that achieves the same error with $\tilde{\boldsymbol{\Phi}}$ containing strictly fewer atoms than $\boldsymbol{\Phi}$ and each column of $\tilde{\mathbf{X}}$ being at least as sparse as the corresponding column of $\mathbf{X}$.*

Proof: If $\mathbf{X}$ does not have full rank, then at least one of its rows lies in the span of the other ones. Let us assume with no loss of generality that the last row $\mathbf{x}^M$ is one such row. Then it can be decomposed as $\mathbf{x}^M = \mathbf{a}\mathbf{X}^{[1,M-1]}$ where $\mathbf{a}$ is a row vector. One can now expand the signal approximation $\boldsymbol{\Phi}\mathbf{X}$:

$$\begin{aligned} \boldsymbol{\Phi}\mathbf{X} &= \boldsymbol{\Phi}_{[1,M-1]}\mathbf{X}^{[1,M-1]} + \boldsymbol{\varphi}_M \mathbf{x}^M && (54) \\ &= \boldsymbol{\Phi}_{[1,M-1]}\mathbf{X}^{[1,M-1]} + \boldsymbol{\varphi}_M \mathbf{a}\mathbf{X}^{[1,M-1]} && (55) \\ &= \left(\boldsymbol{\Phi}_{[1,M-1]} + \boldsymbol{\varphi}_M \mathbf{a}\right)\mathbf{X}^{[1,M-1]} && (56) \\ &= \tilde{\boldsymbol{\Phi}}\tilde{\mathbf{X}} && (57) \end{aligned}$$

The dictionary $\boldsymbol{\Phi}_{[1,m-1]} + \boldsymbol{\varphi}_m \mathbf{a}$ contains $m-1$ atoms and each column of $\mathbf{X}^{[1,m-1]}$ has either $K-1$ or K non-zero coefficients. ■

A strictly better dictionary can then be built using the following method:

- let n be the index of a training signal with a non-zero residual,
- set one its non-zero coefficients $\tilde{x}_n^m$ to 0,
- update the residual $\mathbf{r}_n = \mathbf{y}_n - \tilde{\boldsymbol{\Phi}}\tilde{\mathbf{x}}_n$,
- set $\boldsymbol{\varphi}_M$ to $\frac{\mathbf{r}_n}{\|\mathbf{r}_n\|_2}$ and x_n^M to $\|\mathbf{r}_n\|_2$.

This new point has the same residual as the old one for all training signals except $\mathbf{r}_n$ that is now 0, therefore it is a strictly better point. So the full rank case is the only one that matters in practice: if the algorithm converges towards a solution with rank deficient coefficients, then one should rather build the better dictionary and continue from there.

V. Fixed points of K-SVD

We now investigate the fixed points of K-SVD and how they are related to the fixed points of MOD and Sparsenet. The analysis of the dictionary update is easier for K-SVD because each atom is modified only once per iteration by the update rule (10). Therefore an atom $\boldsymbol{\varphi}_m$ is fixed if and only if it satisfies the condition:

$$\boldsymbol{\varphi}_m = \underset{\mathbf{v}, \|\mathbf{v}\|_2 = 1}{\operatorname{argmax}} \ \mathbf{v}^T \mathbf{E}_{(m)} \mathbf{E}_{(m)}^T \mathbf{v} \tag{58}$$

On the other hand, the coefficients $\mathbf{X}$ are modified twice, once during the coefficient update (16) and once during the dictionary update (11). However the following Lemma shows that this double update is redundant for the fixed point analysis we want to perform.

Lemma 3. *During one iteration of the K-SVD dictionary update, after updating the m^{th} atom, if all the atoms $(\boldsymbol{\varphi}_1, \ldots, \boldsymbol{\varphi}_m)$ updated in the current iteration were left unchanged, then their corresponding coefficients $(\mathbf{x}^1, \ldots, \mathbf{x}^m)$ did not change in the current dictionary update either.*

Proof: The proof is recursive over m. If $m = 1$, then we are updating the first atom. That update happens just after the least-square coefficient update described in Equation (16), so each residual is orthogonal to each atom with a non-zero coefficient. Let us now assume that the atom $\boldsymbol{\varphi}_1$ is fixed for this K-SVD update. After its update, its coefficients $\mathbf{x}^1_{\gamma_1}$ are updated:

$$\begin{aligned} \mathbf{x}^1_{\gamma_1} &\leftarrow \boldsymbol{\varphi}_1^T \mathbf{E}_{(1)} && (59) \\ &= \boldsymbol{\varphi}_1^T [\mathbf{R} + \boldsymbol{\varphi}_1 \mathbf{x}^1]_{\gamma_1} && (60) \\ &= \boldsymbol{\varphi}_1^T \mathbf{R}_{\gamma_1} + \boldsymbol{\varphi}_1^T \boldsymbol{\varphi}_1 \mathbf{x}^1_{\gamma_1} && (61) \\ \mathbf{x}^1_{\gamma_1} &\leftarrow \mathbf{x}^1_{\gamma_1} && (62) \end{aligned}$$

because $\varphi_1^T \mathbf{R}_{\gamma_1} = 0$ and $\varphi_1^T \varphi_1 = 1$. So if the first atom is fixed, then its coefficients do not change either during the dictionary update.

Let us now assume that the Lemma 3 holds for up to m atoms and that the first $m+1$ atoms are fixed for the dictionary update. Since the lemma is true for the first m atoms and they are fixed, then neither the first m atoms nor their coefficients changed during the dictionary update. So the residual $\mathbf{R}_{\gamma_{m+1}}$ is still orthogonal to the atom φ_{m+1} and the same reasoning applied to the first atom φ_1 can be applied to φ_{m+1} to show that its coefficients $\mathbf{x}^{m+1}_{\gamma_{m+1}}$ do not change either. ■

In particular, if the whole dictionary is fixed for the dictionary update, then the coefficients are also fixed for the dictionary update. We can now characterize the fixed points of $\overline{\text{K-SVD}}$.

Theorem 3. *A point* $(\mathbf{\Phi}, \mathbf{X})$ *is a fixed point of* $\overline{\textit{K-SVD}}$ *if and only if it satisfies Condition* (22) *and Condition* (58) *for each atom.*

Proof: Let $(\mathbf{\Phi}, \mathbf{X})$ be a point that satisfies Condtion (22) and Condition (58) for each atom. The point satisfies Condition (22) so its coefficients $\mathbf{X}$ are fixed for the least-square coefficient update. It also satisfies Condition (58) for each atom so the dictionary $\mathbf{\Phi}$ is fixed for the whole K-SVD dictionary update. Because of Lemma 3, that implies that the coefficients $\mathbf{X}$ are also fixed for the dictionary update, thus for the whole $\overline{\text{K-SVD}}$ iteration.

Conversely, let $(\mathbf{\Phi}, \mathbf{X})$ be a fixed point of $\overline{\text{K-SVD}}$. In particular, its dictionary is fixed for the dictionary update, so each atom satisfies Condition (58) and the coefficients $\mathbf{X}$ are also fixed for the dictionary update too because of Lemma 3. So the coefficients at the end of the whole $\overline{\text{K-SVD}}$ iteration are the same as the coefficients after the least-square coefficient update (16) only, and also the same as at the beginning of the iteration (which is also the beginning of the coefficient update) since $(\mathbf{\Phi}, \mathbf{X})$ is a fixed point. So the coefficients are fixed for the coefficient update step, hence they satisfy Condition (22). ■

Although Lemma 3 provides a necessary and sufficient characterizaiton of the fixed of $\overline{\text{K-SVD}}$, Condition (58) is not easy to interpret. The following Lemma shows a link between it and the critical points of the cost function f.

Lemma 4. *If a point satisfies the normalization constraint* (13) *and the fixed point Condition* (22) *for the least-square coefficient update, then it is a critical point of the cost function f if and only if each atom φ_m is a left singular vector of its error matrix* $\mathbf{E}_{(m)}$.

Proof: First of all, the principal component of $\mathbf{E}_{(m)}$ is computed using the Singular Value Decomposition (SVD): it is the left singular vector associated with the highest singular value. It is well known that the left singular vectors of $\mathbf{E}_{(m)}$ are also the eigenvectors of the operator $\mathbf{E}_{(m)}\mathbf{E}_{(m)}^T$. Let $(\mathbf{\Phi}, \mathbf{X})$ be a critical point of f. It satisfies the conditions (20) and (21). Let φ_m be the m^{th} atom of $\mathbf{\Phi}$, $\mathbf{E}_{(m)} = \mathbf{R}_{\gamma^m} + \varphi_m \mathbf{x}^m_{\gamma^m}$ the corresponding restricted error and γ^m the corresponding co-support. Then

$$\mathbf{E}_{(m)}\mathbf{E}_{(m)}^T \varphi_m = \mathbf{E}_{(m)} \left(\mathbf{R}_{\gamma^m} + \varphi_m \mathbf{x}^m_{\gamma^m}\right)^T \varphi_m \tag{63}$$

$$= \mathbf{E}_{(m)} \left(\mathbf{R}^T_{\gamma^m} \varphi_m + \mathbf{x}^m_{\gamma^m}{}^T \varphi_m^T \varphi_m\right) . \tag{64}$$

Thanks to Condition (21) and the normalization of φ_m, we can simplify further:

$$\mathbf{E}_{(m)}\mathbf{E}_{(m)}^T \varphi_m = \mathbf{E}_{(m)} \left(0 + \mathbf{x}^m_{\gamma^m}{}^T\right) \tag{65}$$

$$= \left(\mathbf{R}_{\gamma^m} + \varphi_m \mathbf{x}^m_{\gamma^m}\right) \mathbf{x}^m_{\gamma^m}{}^T \tag{66}$$

$$= \mathbf{R}_{\gamma^m} \mathbf{x}^m_{\gamma^m}{}^T + \varphi_m \mathbf{x}^m_{\gamma^m} \mathbf{x}^m_{\gamma^m}{}^T \tag{67}$$

$$= 0 + \varphi_m \mathbf{x}^m_{\gamma^m} \mathbf{x}^m_{\gamma^m}{}^T \tag{68}$$

$$= \varphi_m \left\|\mathbf{x}^m_{\gamma^m}\right\|_2^2 \tag{69}$$

Due to Equation (20). This proves that for any critical point of f, each atom of the dictionary is an eigenvector of the frame operator $\mathbf{E}_{(m)}\mathbf{E}_{(m)}^T$ of its restricted error matrix, with the energy of its coefficients as the associated eigenvalue.

Conversely, let us assume that the fixed point condition for OMP (22) is satisfied and that an atom φ_m is an eigenvector of the frame operator $\mathbf{E}_{(m)}\mathbf{E}_{(m)}^T$ with eigenvalue λ:

$$\mathbf{E}_{(m)}\mathbf{E}_{(m)}^T \varphi_m = \lambda \varphi_m . \tag{70}$$

The value of the eigenvalue λ can be computed as follows:

$$\mathbf{E}_{(m)}^T \varphi_m = \left(\mathbf{R}_{\gamma^m} + \varphi_m \mathbf{x}^m_{\gamma^m}\right)^T \varphi_m \tag{71}$$

$$= \mathbf{R}^T_{\gamma^m} \varphi_m + \mathbf{x}^m_{\gamma^m}{}^T \varphi_m^T \varphi_m \tag{72}$$

$$= 0 + \mathbf{x}^m_{\gamma^m}{}^T = \mathbf{x}^m_{\gamma^m}{}^T \tag{73}$$

$$\lambda = \varphi_m^T \mathbf{E}_{(m)}\mathbf{E}_{(m)}^T \varphi_m \tag{74}$$

$$= \mathbf{x}^m_{\gamma^m} \mathbf{x}^m_{\gamma^m}{}^T = \left\|\mathbf{x}^m_{\gamma^m}\right\|_2^2 . \tag{75}$$

Thus the eigenvalue λ is the energy of the coefficients. Now let us expand the Equation (70)

$$\mathbf{E}_{(m)}\mathbf{E}_{(m)}^T \varphi_m = \lambda \varphi_m \tag{76}$$

$$\mathbf{E}_{(m)} \mathbf{x}^m_{\gamma^m}{}^T = \varphi_m \left\|\mathbf{x}^m_{\gamma^m}\right\|_2^2 \tag{77}$$

$$\mathbf{R}_{\gamma^m} \mathbf{x}^m_{\gamma^m}{}^T + \varphi_m \mathbf{x}^m_{\gamma^m} \mathbf{x}^m_{\gamma^m}{}^T = \varphi_m \mathbf{x}^m_{\gamma^m} \mathbf{x}^m_{\gamma^m}{}^T \tag{78}$$

$$\mathbf{R}_{\gamma^m} \mathbf{x}^m_{\gamma^m}{}^T = 0 . \tag{79}$$

We notice the similarity between Equations (79) and (20). There are two differences between them. First, Equation (20) uses the whole coefficient matrix $\mathbf{X}$ instead of just a row $\mathbf{x}^m$. This is just a more compact way of writing

$$\mathbf{R}\mathbf{x}^{mT} = 0, \ \forall m \in [1, M]. \tag{80}$$

More important, Equation (79) is restricted to the co-support γ^m. By definition of the co-support, its complementary $\overline{\gamma^m}$ contains only indices such that $\mathbf{x}^m_{\overline{\gamma^m}} = 0$. Therefore the left hand sides of Equations (79) and (80) are equal, thus the equations are equivalent. So the critical point condition (20) is equivalent to having all the atoms satisfying Condition (79). If all the atoms are singular vectors of their restricted errors (thus all satisfying Condition (79)) and the decomposition is orthogonal, then the point $(\mathbf{\Phi}, \mathbf{X})$ is a critical point. ■

We now have all the elements to prove that the K-SVD stability condition is stricter than the critical point condition, thus the MOD and Sparsenet stability conditions.

Theorem 4. *Every fixed point of K-SVD is also a fixed point of MOD and Sparsenet. The opposite is not true in general.*

Proof: The fixed points of MOD and Sparsenet are the critical points of the cost function f. Lemma 4 shows that the critical points of f are the points where each atom is *a* left singular vector of its restricted error matrix associated with *any* singular value. The fixed points of K-SVD are the points where each atom is *the* left singular vector of its restricted error matrix associated with *the largest* singular value value. So MOD and Sparsenet can have more fixed points than K-SVD whenever at least one of the restricted error matrices has more than one singular value, i.e. the signal dimension D is at least 2 and at least one of the atoms is used in the representation of at least 2 signals.

If an atom is an eigenvector not associated with the highest eigenvalue, it is a fixed point for MOD and Sparsenet but not for K-SVD. This can easily be done in dimension $D = 2$ with a dictionary containing only $M = 1$ atom and $N = 2$ training signals. Let us consider the following training data and point:

$$\mathbf{Y} = \begin{pmatrix} 2 & 2 \\ -1 & 1 \end{pmatrix} \qquad \mathbf{\Phi} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \qquad \mathbf{X} = \begin{pmatrix} -1 & 1 \end{pmatrix}.$$

For the Sparsenet update, the gradient is given by:

$$\mathbf{R} = \begin{pmatrix} 2 & 2 \\ 0 & 0 \end{pmatrix} \qquad \frac{\partial f}{\partial \mathbf{\Phi}} = \mathbf{R}\mathbf{X}^T = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \ .$$

The gradient is 0 so the point is fixed for Sparsenet.

Of course the theory tells us that this point is also fixed for MOD, but let us develop the MOD iteration anyway:

$$\mathbf{X}^+ = \mathbf{X}^T \left(\mathbf{X}\mathbf{X}^T\right)^{-1} = \begin{pmatrix} -\frac{1}{2} \\ \frac{1}{2} \end{pmatrix}$$

$$\mathbf{Y}\mathbf{X}^+ = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \mathbf{\Phi}$$

Both Sparsenet and MOD would stop at this point with an error energy $\|\mathbf{R}\|_F^2 = 8$.

For K-SVD, there is only one atom in the dictionary so the restricted error $\mathbf{E}$ is the same as the orginal signal $\mathbf{Y}$ and the frame operator $\mathbf{E}\mathbf{E}^T$ is equal to:

$$\mathbf{E}\mathbf{E}^T = \begin{pmatrix} 8 & 0 \\ 0 & 2 \end{pmatrix}$$

which is already in diagonal form. Its two eigenvalues are 8 and 2. One can notice that the current dictionary is the eigenvalue associated with the eigenvalue 2. In the next step K-SVD would choose the eigenvecotr associated with 8 instead:

$$\mathbf{\Phi} \leftarrow \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \mathbf{X} \leftarrow \begin{pmatrix} 2 & 2 \end{pmatrix} \qquad \mathbf{R} = \begin{pmatrix} 0 & 0 \\ -1 & 1 \end{pmatrix}$$

which would result in an error of 2 only. ■

VI. Experimental results

We now investigate whether in practice, the points towards which Sparsenet and MOD converge are also fixed points for K-SVD. In that goal we tried to run first MOD and Sparsenet, then K-SVD. All these experiments were performed using the SMALLBox toolbox [12][1].

These experiments were performed on noiseless synthetic data in dimension $D = 64$. We generated 1200 training sets of sparse signals over the union of a Dirac and a Discrete Fourier Transform bases. Each set contains $N = 256$ training signals of sparsity $K = 8$ with uniform i.i.d. supports and Gaussian i.i.d. values for the non-zero amplitudes. Those dimensions were tuned by hand using K-SVD as a reference so that so problem is neither too easy (a global minimum is always reached) nor too difficult (a global minimum is never reached).

A. Learning with a known support

For the first experiments, we set the support used for the synthesis of the signals as the given. Under these conditions there is potential for the algorithms to converge towards a global minimum and that can be detected when the error decreases to 0. We first

[1] https://code.soundsoftware.ac.uk/projects/smallbox

ran each of the three algorithms for 2000 iterations, then after each of them we ran K-SVD for another 2000 iterations starting from the point reached by each algorithm. The executions were initialized with a random Gaussian dictionary.

The performance of each algorithm is measured with the approximation SNR defined as

$$SNR = -10 \log_{10} \frac{\|\mathbf{R}\|_F^2}{\|\mathbf{Y}\|_F^2}. \tag{81}$$

The global minimum of the error is 0 so the SNR can increase to infinity. In practice the SNR does not increase above ≈300dB due to numerical precision on our machine. The exact value of that threshold and the amplitude of the oscillations generated above it depend both on the algorithm and the implementation so we set the decision criterion lower than 300dB: we consider that an execution has successfully reached a global minimum if the SNR reaches 250dB. Table I shows the exact recovery rates achieved by each algorithm after 2000 iterations and after running K-SVD for another 2000 iterations.

K-SVD alone succeeds in recovering the best dictionary in 20% of the cases. MOD only succeeds in 4% of the cases and the succession of MOD and K-SVD does not significantly improve the results. Sparsenet alone only succeeds in 10 cases out of 1200, but the combination of Sparsenet and K-SVD recovers the best dictionary 98% of the cases. This is a major improvement over any of the 3 studied algorithms.

Figure 1 shows the typical evolution of the SNR over time for the different algorithms and some representative training sets. The chosen training sets are the 11 quantiles for the area under the SNR curve, i.e. the sets with the best and worst overall SNR, and every 10% in between, plus a few handpicked experiments that exhibit interesting dynamic properties. The actual recovery rates are presented in Table I. On the Sparsenet plot we can see a sharp acceleration at 2000 iterations, when we switch from Sparsenet to K-SVD. This acceleration is not observed between MOD and K-SVD, suggesting that MOD converges in practice towards points that are also fixed for K-SVD.

Besides these plots provide some hindsight about the dynamic behavior of the algorithms. MOD presents a strong tendency to evolve by jumps rather than steady increases. When K-SVD reaches the global minimum, convergence is quite fast: numerical precision is often reached in less than 500 iterations. However, when it does not find a global minimum, convergence is much slower: the SNR keeps steadily decreasing, albeit at an very slow rate. We ran longer experiments for a few of the data and the numerical precision was still not reached after 40000 iterations. Besides, we also observed in some rare cases that the SNR can remain almost stationary for a very long time before the algorithm finally finds a global optimum. This was observed both for MOD and K-SVD. So testing whether a dictionary learning algorithm has converged or not seems to be a hard problem on its own.

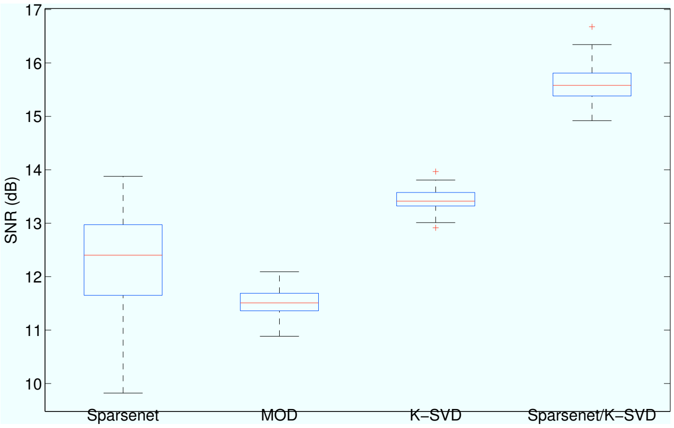


Figure 2. Repartition of the final SNR over the 100 different runs for Sparsenet, MOD, K-SVD and Sparsenet/K-SVD. The box contains 50% of the distribution with the median marked inside and the whiskers the whole distribution. Sparsenet/K-SVD performs much better than of the other 3 algorithms taken alone, including K-SVD.

B. Complete learning, including the sparse decomposition step

We also ran Sparsenet, MOD, K-SVD and Sparsenet followed by K-SVD on a subset of the same data (only 100 training sets), but this time without the use of a known support. The algorithms were initialized with a random 8-sparse coefficient matrix X then started with the dictionary update. The observed behavior was homogeneous among all different algorithms and data: the SNR first improves quickly, then slows down and finally suffers from small random variations when OMP starts to fail: at good SNRs, OMP can compute a decomposition with an error higher than with the previous iteration's decomposition.

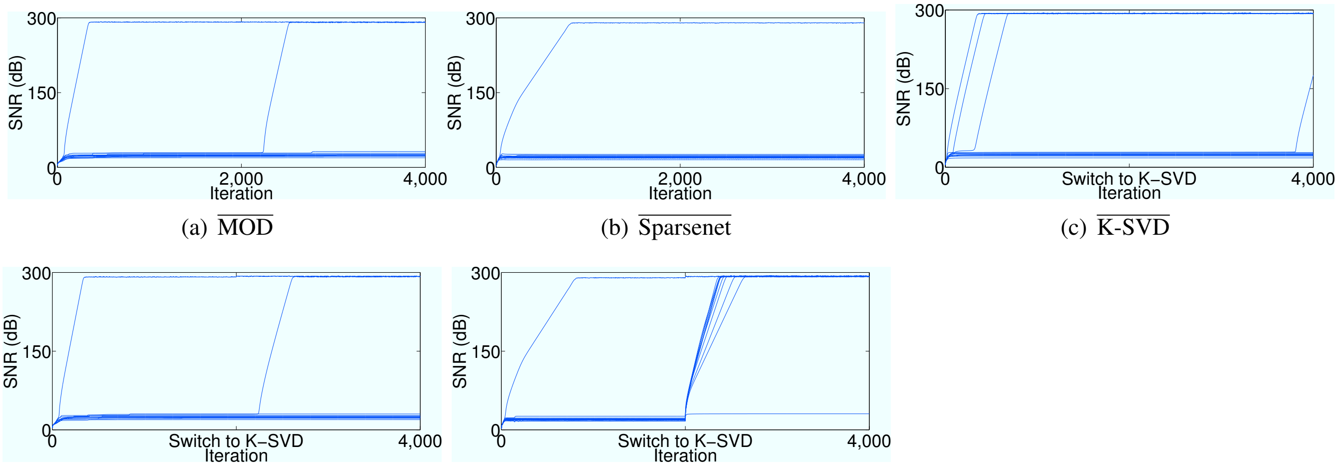
Figure 2 shows the distribution of the SNR reached after 500 iterations. Sparsenet has a much higher variability than the other algorithms. However Sparsenet followed by K-SVD significantly outperforms any of the other 3 algorithms, including K-SVD taken alone. The average improvement over K-SVD is 2dB.

VII. Conclusion

Dictionary learning is a complex problem. In this work we considered the simplified problem where the

Table I
EXACT RECOVERY RATES FOR THE DIFFERENT ALGORITHM COMBINATIONS

Algorithm	One algorithm for 4000 iterations	One algorithm for 2000 iterations then K-SVD
$\overline{\text{K-SVD}}$	20%	20%
$\overline{\text{MOD}}$	4%	4%
$\overline{\text{Sparsenet}}$	1%	98%



(a) $\overline{\text{MOD}}$ (b) $\overline{\text{Sparsenet}}$ (c) $\overline{\text{K-SVD}}$

(d) $\overline{\text{MOD}}$ then $\overline{\text{K-SVD}}$ (e) $\overline{\text{Sparsenet}}$ then $\overline{\text{K-SVD}}$

Figure 1. Some typical plots of the SNR depending on the iteration when running any algorithm for 4000 iterations (upper row), and when running MOD or Sparsenet for 2000 iterations, then K-SVD for 2000 iterations (lower row).

supports of the decomposition is known in advance. Even in that case, the remaining optimization problem is still non-convex, and we observed that existing algorithms typically fail to solve it. We also proved that K-SVD has strictly fewer fixed points than MOD and Sparsenet. Finally we found that running Sparsenet followed by K-SVD yields substantially better results than any algorithm run alone, and confirmed those results when solving the actual dictionary learning problem when the support is unknown.

The proposed results also raise several other questions. First, there is no clear link so far between the set of the local minima of f and the set of the fixed points for K-SVD. The theory developed here also does not explain the observed difference of behavior between MOD and Sparsenet. It may be that some critical points of the cost function are unstable, and that those unstable points correspond to the steps we observed in the progression of MOD. On the other hand, since Sparsenet uses a fixed descent stepsize, it converges much more slowly, if at all. Despite showing similar performance to MOD and K-SVD, Sparsenet could compute points that are different in nature, non-stationary points in the attraction basin of the global optimum rather than local optima or other suboptimal stationary points. We have started investigating that idea [13] and intend to continue that work.

REFERENCES

[1] E. Ravelli, G. Richard, and L. Daudet, “Union of MDCT bases for audio coding,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1361–1372, Nov. 2008.

[2] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[3] J. Mairal, M. Elad, and G. Sapiro, “Sparse representation for color image restoration,” *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, Jan 2008.

[4] A. Adler, V. Emiya, M. Jafari, M. Elad, R. Gribonval, and M. D. Plumbley, “A constrained matching pursuit approach to audio declipping,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2011, pp. 329–332.

[5] B. A. Olshausen and D. J. Field, “Emergence of simple-cell receptive field properties by learning a sparse code for natural images,” *Nature*, vol. 381, pp. 607–609, Jun 1996.

[6] K. Engan, S. Aase, and J. Hakon Husoy, “Method of optimal directions for frame design,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 5, 1999, pp. 2443–2446 vol.5.

[7] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, Nov 2006.

[8] S. Mallat and Z. Zhang, “Matching pursuits with time-frequency dictionaries,” *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, Dec 1993.

[9] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, “Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition,” in *Conference Record of The*

Twenty-Seventh Asilomar Conference on Signals, Systems and Computers, 1993., Nov. 1993, pp. 40–44 vol.1.

[10] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[11] B. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, pp. 227–234, 1995.

[12] I. Damnjanovic, M. E. P. Davies, and M. D. Plumbley, "SMALLbox - an evaluation framework for sparse representations and dictionary learning algorithms." in *Latent Variable Analysis (LVA/ICA)*, 2010, pp. 418–425.

[13] B. Mailhé and M. D. Plumbley, "Dictionary learning with large step gradient descent for sparse representations," in *Latent Variable Analysis (LVA/ICA)*, 2012, pp. 231–238.