Dictionary Learning for Sparse Approximations with the Majorization Method

Mehrdad Yaghoobi, *Member, IEEE,* and Thomas Blumensath, *Member, IEEE* and Mike E. Davies, *Member, IEEE*

***Abstract*—In order to find sparse approximations of signals, an appropriate generative model for the signal class has to be known. If the model is unknown, it can be adapted using a set of training samples. This paper presents a novel method for dictionary learning and extends the learning problem by introducing different constraints on the dictionary. The convergence of the proposed method to a fixed point is guaranteed, unless the accumulation points form a continuum. This holds for different sparsity measures. The majorization method is an optimization method that substitutes the original objective function with a surrogate function that is updated in each optimization step. This method has been used successfully in sparse approximation and statistical estimation (e.g. Expectation Maximization (EM)) problems. This paper shows that the majorization method can be used for the dictionary learning problem too. The proposed method is compared with other methods on both synthetic and real data and different constraints on the dictionary are compared. Simulations show the advantages of the proposed method over other currently available dictionary learning methods not only in terms of average performance but also in terms of computation time.**

***Index Terms*—Dictionary Learning, Sparse Approximation, Majorization Methods, Surrogate Function Optimization Method, Block Relaxation Methods, Constrained Optimization**

I. Introduction

ORTHOGONAL function representations, introduced in the nineteenth century, are still a powerful tool in signal analysis. These representations have unique characteristics that make them suitable for many signal processing applications. In the last two decades, many researchers have tried to extend this idea to non-orthogonal and overcomplete representations [1], [2]. The overcomplete representation problem with the associated underdetermined linear system does not have a unique solution. The method of frames finds the minimum mean square solution and leads to representations where most of the coefficients are non-zero. Minimum mean square representations are desirable for some applications (e.g. robust transform coding in the presence of noise or erasure [3]) while there are other applications where sparsity of the representation is more desirable, e.g. in Compressed Sensing [4].

Let $\mathbf{y} \in \mathbb{R}^d$ and $\mathbf{x} \in \mathbb{R}^N$ be the input signal and the coefficient vector respectively. The sparsest representation would be,

$$\min_x ||\mathbf{x}||_0 \quad s.t \quad \mathbf{y} = \mathbf{D}\mathbf{x}, \tag{1}$$

where $\mathbf{D}$ is a $d \times N$ matrix, often called *dictionary* and $||.||_0$ is the sparsity measure that counts the number of non-zero coefficients. This formulation can be relaxed to sparse *approximations* by using $||\mathbf{y} - \mathbf{D}\mathbf{x}||_2 \leq \epsilon$ with a small constant ϵ. Unfortunately finding the solutions to the above combinatorial problems is not easy in general [5]. Many approximations/relaxations have been presented to find acceptable solutions, e.g. [6], [7].

These methods are more successful at finding a sparse $\mathbf{x}$, when there is a suitable dictionary for the given signal. A simple method for dictionary generation is to add two or more orthogonal bases. Block-wise orthogonality can then be exploited to find the sparse approximation [8]. This also makes it easier to analyze the performance of sparse approximation methods [9], [10]. Another way to design a dictionary is to sample the parameters of an analytic function. For example a famous dictionary that has been used for overcomplete audio and image representations, is the Gabor dictionary [6]. These designed dictionaries are efficient when we have some a priori information about the signal's generative model. Alternatively it is possible to adapt the dictionary to a given source using a set of training samples ($\mathscr{Y} = \{\mathbf{y}^{(i)} : 1 \leq i \leq L\}$ where L is the number of training samples). Dictionary learning is the process of finding a dictionary $\mathbf{D}$ in which a given set of training samples has sparse representations (or approximations) $\mathscr{X} = \{\mathbf{x}^{(i)} : 1 \leq i \leq L\}$. Different methods have been proposed to learn dictionaries [11]–[15]. These methods are generally based on alternating minimization. In one step, a sparse approximation/representation algorithm finds sparse representations of the training samples with a fixed dictionary. In the other step, the dictionary $\mathbf{D}$ is updated to decrease the average approximation error while $\mathscr{X}$ (or the sparsity of $\mathscr{X}$ [15]) remains fixed. Because the objective functions are non-convex based on the pair of parameters $(\mathbf{D},\mathscr{X})$, these methods generally only find a local minimum and different initial value for $\mathbf{D}$ (or $\mathscr{X}$), lead to different solutions. Nevertheless, in practice, good results have been reported [16], [17]. The proposed method in this paper uses a general formulation of alternating minimization. Therefore like other methods, we only expect to find local minima in general.

The authors are with the Institute for Digital Communication and with the Joint Research Institute for Signal and Image Processing, Edinburgh University, Kings Buildings, Mayfield Road, Edinburgh EH9 3JL, UK (e-mail: yaghoobi@ieee.org, thomas.blumensath@ed.ac.uk, mike.davies@ed.ac.uk).

This research was supported by EPSRC grant D000246/1. MED acknowledges support of his position from the Scottish Funding Council and their support of the Joint Research Institute with the Heriot-Watt University as a component part of the Edinburgh Research Partnership.

Contributions of the paper

This paper introduces a new algorithm for constrained dictionary learning which is very flexible and can use different constraints on the dictionary. The given method uses convex admissible sets whose boundaries are the same as the most frequently used admissible sets, however these convex sets allow the algorithm to generate a sequence throughout the sets (and not only on their boundaries). An advantage of the given method is that it optimizes a joint parameter objective function of the sparse coefficient matrix and the dictionary. In this framework, it is possible to choose a better path from the initial to the learnt dictionary by reducing the objective in different directions (coefficients or dictionary) in a cyclic way. This prevents oscillations of the sequence of updates around the optimal path and makes the algorithm more suitable for large scale problems, for which the calculation of sparse approximations of the training samples is often impossible.

Another advantage of the proposed algorithm is that we can impose a tighter constraint on the dictionary. For example, when a minimum size dictionary is required or when the optimum size of the dictionary is unknown, we can impose an additional penalty on the number of the atoms in the dictionary.

Numerical results show that the algorithm is faster than (or at least as fast as) most of the available dictionary learning methods.

Finally we show that the new algorithm is not only stable but also converges to a fixed point or its accumulation points form a continuum (in contrast to most of the dictionary learning methods, for which so far only stability has been shown).

Organization of the paper

An overview of previous dictionary learning methods is presented in Section II. Section III introduces the dictionary learning framework used in this paper. We introduce two new admissible sets for the dictionaries. Then, in Section III-A, we introduce the majorization method which is used in the matrix valued sparse approximation (III-B) and the dictionary update (III-C1, III-C2) steps. We introduce a new objective function to penalize the size of the dictionary in Section III-D. By minimization of the new objective function with the majorization minimization method, we find a minimum size dictionary. The different dictionary update methods are examined in the simulation section using training samples generated synthetically or sampled from an audio signal. After concluding the paper we present a convergence proof of the algorithm in Appendix B.

Notation

In this paper we use the following conventions. We use small and capital bold face characters for vector and matrix valued parameters respectively. In an iterative algorithm, the value of a parameter in the k^{th} iteration is distinguished by using the iteration number in square brackets, e.g. $\mathbf{D}^{[k]}$. We use a similar notation for a countable series. When a parameter appears with a hat, it shows the current value of that parameter. In the majorization method we introduce an auxiliary parameter which is distinguished with a double dagger superscript, e.g. $\mathbf{X}^{\ddagger}$. In dictionary learning, we have a set of training signals $\mathbf{y}^{(i)}$, where i is the signal index. Similarly, the associated coefficient vectors are $\mathbf{x}^{(i)}$. In this paper we use different norms for vectors and matrices. $||.||$ and $||.||_F$ are spectral and Frobenius norm in the Euclidean vector space respectively. $||.||_p : 0 < p \leq 1$ is the ℓ_p quasi-norm $(\sum |.|^p)^{\frac{1}{p}}$.

II. Dictionary Learning Methods

In traditional dictionary learning, one often starts with some initial dictionary and finds sparse approximations of the set of training signals while keeping the dictionary fixed. This is followed by a second step in which the sparse coefficients are kept fixed and the dictionary is optimized. This algorithm runs for a specific number of alternating optimizations or until a specific approximation error is reached. Most of these algorithms have been derived for dictionary learning in a noisy sparse *approximation* setting. Recently some researchers have considered dictionary learning for exact sparse representations [18], [19]. Like most other researchers, we consider dictionary learning for sparse approximation.

A. Sparse Approximation

Given a set of training samples $\mathbf{y}^{(i)}, \forall i: \ 1 \leq i \leq L$ and a dictionary $\mathbf{D}$, sparse approximations are often found by [1],

$$\begin{aligned} \mathbf{x}^{(i)^*} &= \arg\min_{\mathbf{x}^{(i)}} \phi_i(\mathbf{x}^{(i)}) \ ; \\ \phi_i(\mathbf{x}) &= ||\mathbf{y}^{(i)} - \mathbf{D}\mathbf{x}||^2 + \lambda ||\mathbf{x}||_p^p \ , \ p \leq 1 \end{aligned} \tag{2}$$

An alternative to minimizing (2) individually on each vector is to find a joint sparse approximation of the matrix $\mathbf{Y} = [\mathbf{y}^{(1)} \ \mathbf{y}^{(2)} \ ... \ \mathbf{y}^{(L)}]$ by employing a sparsity measure in matrix form. The sparse matrix approximation problem can be formulated as,

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} \phi(\mathbf{X}) \ ; \ \phi(\mathbf{X}) = ||\mathbf{Y} - \mathbf{D}\mathbf{X}||_F^2 + \lambda J_{p,p}(\mathbf{X}), \tag{3}$$

where $J_{p,q}(\mathbf{X})$ is defined as [20],

$$J_{p,q}(\mathbf{X}) = \sum_{i \in I} [\sum_{j \in J} |x_{ij}|^q]^{p/q}. \tag{4}$$

For example, $||\mathbf{X}||_F = J_{2,2}^{1/2}(\mathbf{X})$ would be the Frobenius-norm. When $p = q$ all elements in $\mathbf{X}$ are treated equally.

In this paper we use $p = 1$, so that $J_{p,p}$ is convex. Extending the algorithm to $0 < p < 1$ is possible by using the majorization method proposed in [21]. However the convergence of the algorithm in this setting has not yet been proven [21], [22].

[1] Instead of minimizing an objective function like (2) one can also use a greedy algorithm. Because greedy algorithms do not deal with an objective function explicitly, convergence analysis of dictionary learning based on these methods is not easy and is therefore not considered here.

B. Dictionary Update

The second step in dictionary learning is the optimization of the dictionary based on the current sparse approximation. The cost function in (3) can be thought of as an objective function with two parameters,

$$\phi(\mathbf{D},\mathbf{X}) = ||\mathbf{Y}-\mathbf{D}\mathbf{X}||_F^2 + \lambda J_{1,1}(\mathbf{X}). \tag{5}$$

Without additional constraints on the dictionary, minimizing the above objective function is an ill-posed problem. By constraining the norm of $\mathbf{D}$ we can solve the scale ambiguity[2] of the problem. Dictionaries with fixed column-norms or fixed Frobenius-norm have been used in different papers (for example [13] and [23]). We will use more general convex admissible sets defined in (7) and (8) below.

C. Previously Suggested Dictionary Update Methods

In the Method of Optimal Directions (MOD) [13] the best dictionary D is found using the pseudo inverse of X, followed by re-normalization of each atom. The Maximum Likelihood based Dictionary Learning algorithm [11], is similar to MOD but uses gradient optimization. In general, if the update is done iteratively, the best possible dictionary is typically calculated without any constraint. This update is then followed by normalization of the atoms. This normalization step can increase the total approximation error.

Kreutz-Delgado et al. [23] presented a dictionary learning method based on Maximum *a Posteriori* estimation (from now called MAP-DL[3]). By the use of an iterative method they estimate a dictionary that is consistent with a Bayesian model [23]. However, as reported in [15], when a fixed column-norm constraint is used, the algorithm updates atom by atom, making the method too slow for many applications.

The K-SVD method presented in [15] is fundamentally different from these methods. Instead of keeping the sparse coefficients fixed in the dictionary update step, only the support of the coefficient vectors (the positions of the non-zero coefficients) is kept fixed. Updates for each atom are found as the best normalized elementary function that matches the error (calculated after representing the signals with all atoms except the currently selected atom).

The formulation of the problem in this paper has several similarities with MAP-DL. However, our approach to solve this problem is based on a joint objective function for both the sparse approximation and the dictionary, which is good because we can develop a uniform approach for the updates and we have the flexibility to be able to switch between updating parameters easily. Furthermore, we use a different class of constraints on the desired dictionaries. In this setting, we will show a basic convergence proof. Our simulations furthermore show faster convergence for the proposed approach. Moreover, we can optimize the joint parameter objective function more wisely (see section III-E) and thereby increase the observed speed of convergence even further.

III. Dictionary Learning with the Majorization Method

We consider the dictionary learning problem as the following constrained optimization problem,

$$\begin{aligned} &\min_{\mathbf{D},\mathbf{X}} \phi(\mathbf{D},\mathbf{X}) \ s.t.\ \mathbf{D} \in \mathscr{D} \\ &\phi(\mathbf{D},\mathbf{X}) = ||\mathbf{Y}-\mathbf{D}\mathbf{X}||_F^2 + \lambda J_{p,p}(\mathbf{X}), \end{aligned} \tag{6}$$

where $\mathscr{D}$ is an admissible set of dictionaries. As noted in [23], two typical constraints are the unit Frobenius-norm and the unit column-norm constraints, both of which lead to non-convex solution sets. Instead of using these constraints in the algorithm derived below, we use the convex relaxed version of these constrained sets. These are the convex sets of matrices with bounded Frobenius norm,

$$\mathscr{D}_F = \{\mathbf{D}_{d\times N} : ||\mathbf{D}||_F \le c_F^{1/2}\} \tag{7}$$

where c_F is a constant and the convex set of matrices with bounded column norm,

$$\mathscr{D}_C = \{\mathbf{D}_{d\times N} : ||\mathbf{d}_i||_2 \le c_C^{1/2}\}, \tag{8}$$

where $\mathbf{d}_i$ is the i$^{\text{th}}$ column of the dictionary $\mathbf{D}$ and c_C is a constant. Note that when the sparsity measure in the sparse approximation step penalizes coefficients based on their magnitudes (e.g. $l_p : 0 < p \le 1$), it is easy to show that the solution of (6) is on the boundary of these convex admissible sets. However, the convex admissible sets also allow the optimization algorithm to "pass through" these admissible sets while the traditional non-convex sets only allow the algorithm to move along the boundary of these sets.

We use the block relaxation technique (see for example [24]) to solve (6), where $p = 1$, that is, in one step we fix $\mathbf{D}$ and minimize the objective based on $\mathbf{X}$, while in the other step we minimize the objective based on $\mathbf{D}$ with $\mathbf{X}$ fixed. This alternating minimization continues until the algorithm converges to an accumulation point. For a fixed dictionary, ℓ_1 penalized sparse approximation is a convex optimization problem and using convex dictionary admissible sets also turns the dictionary update into a convex optimization problem. Whilst this allows us to find the optimum update in each step, (5) is not convex as a function of the pair $(\mathbf{X},\mathbf{D})$, and alternating optimization is not guaranteed to find a global optimum.

Various methods have been presented to solve the ℓ_1 penalized sparse approximation [7], [25], [26]. We choose an Iterative Thresholding (IT) approach, which is a majorization minimization algorithm (see next subsection), which can be extended to the sparse approximation problem in matrix form (see III-B).

A. Majorization Minimization Method

Optimization of the problem in (6) with respect to any one of the parameters is challenging. We here use a technique called the "majorization method" [24], [27]. In the

[2] Approximation error does not change by scaling up one parameter and scaling down the other one with the same scaling factor. Therefore the optimum $\mathbf{X}$ and $\mathbf{D}$ tend to zero and infinity respectively to minimize the sparsity penalty.

[3] Although MAP actually refers to an objective, MAP-DL is an algorithm for dictionary learning based on the MAP objective.

majorization method, the objective function is replaced by a surrogate objective function which majorizes it and can be easily minimized. Here we are particularly interested in surrogate functions in which the parameters are decoupled, so that the surrogate function can be minimized element-wise.

A function ψ majorizes ϕ when it satisfies the following conditions,

$$\begin{aligned} \phi(\omega) &\leq \psi(\omega,\xi),\ \forall\omega,\xi \in \Upsilon \\ \phi(\omega) &= \psi(\omega,\omega),\ \forall\omega \in \Upsilon, \end{aligned} \tag{9}$$

where Υ is the parameter space. The surrogate function has an additional parameter ξ. At each iteration we first choose this parameter as the current value of ω and find the optimal update for ω.

$$\omega_{new} = \arg\min_{\omega\in\Upsilon} \psi(\omega,\xi) \tag{10}$$

We then update ξ with ω_{new}. The algorithm continues until we find an accumulation point. In practice the algorithm is terminated when the distance between ω and ω_{new} is less than some threshold.

This iterative method can be viewed as a block-relaxed minimization of the joint objective $\psi(\omega,\xi)$ [24]. In one step, we find the minimum of ψ based on ω. In the next step we minimize the objective based on ξ.

$$\xi_{new} = \arg\min_{\xi\in\Upsilon} \psi(\omega,\xi) \tag{11}$$

In our formulation, minimization of $\psi(\omega,\xi)$ based on ξ is done using $\xi_{new} = \omega$ (due to the definition of majorization in (9)). We use this interpretation of the majorization method to show the convergence of the proposed method in Appendix B.

There are different ways to derive a surrogate function. Jensen's inequality and Taylor series have often been used for this purpose [28] [29]. The Taylor series of a differentiable function $\phi(\omega)$ is,

$$\phi(\omega) = \phi(\xi) + d\phi(\xi)(\omega-\xi) + \frac{1}{2!}d^2\phi(\xi)(\omega-\xi)^2 + o(\omega^3). \tag{12}$$

When ϕ has a bounded curvature ($d^2\phi < c_s$ for a finite constant c_s) this is majorized by,

$$\phi(\omega) \leq \phi(\xi) + d\phi(\xi)(\omega-\xi) + \frac{c_s}{2}(\omega-\xi)^2, \forall\omega,\xi\in\Omega, \tag{13}$$

and we can define $\psi(\omega,\xi)$ (which satisfies (9)) as follows,

$$\psi(\omega,\xi) = \phi(\xi) + d\phi(\xi)(\omega-\xi) + \frac{c_s}{2}(\omega-\xi)^2. \tag{14}$$

Then, at each iteration, $\phi(\omega_{new}) \leq \psi(\omega_{new},\omega) \leq \psi(\omega,\omega) = \phi(\omega)$, hence ϕ does not increase. Conditions for which these algorithms converge have been presented in [24] and [29]. The convergence of this method for sparse approximation is shown in [26]. A similar analysis can be derived for the iterative method in the dictionary update step.

In the next sections we show how we can use the majorization method to optimize the objective introduced in (6) based on $\mathbf{X}$ (Section III-B) or $\mathbf{D}$ (Sections III-C and III-D) using different constraints. Updating the coefficient or the dictionary matrices always reduces the joint objective function or keeps it at the same value. The fact that the objective function is lower-bounded is sufficient to show stability of the updating process in the sense of Lyapunov (Lyapunov second theorem) [30]. We also provide a basic convergence proof for the proposed algorithm in Appendix B.

Algorithm 1 : $\mathcal{SA}(\mathbf{X}_t, \mathbf{D}_t)$

```
initialization: $c_X > \|\mathbf{D}_t^T\mathbf{D}_t\|$ , $\mathbf{X}^{[0]} = \mathbf{X}_t$
for $n = 1$ to $K_X$ do
    $\mathbf{A} = \frac{1}{c_X}(\mathbf{D}_t^T\mathbf{Y} + (c_X\mathbf{I} - \mathbf{D}_t^T\mathbf{D}_t)\mathbf{X}^{[n-1]})$
    $\mathbf{X}^{[n]} = \mathcal{S}_\lambda(\mathbf{A})$
end for
output: $\mathbf{X}_{t+1} = \mathbf{X}^{[K_X]}$
```

B. Matrix Valued Sparse Approximation

We begin by showing how the majorization method is used for the first step of the alternating minimization: matrix valued sparse approximation. The updating formula derived here is used in the generalized block relaxation method derived later in this section. For fixed $\mathbf{D}$, we use the matrix form of the Taylor series inequality (13), see Appendix A, to derive the following majorizing function,

$$\begin{aligned} ||\mathbf{Y}-\mathbf{DX}||_F^2 \leq\ & ||\mathbf{Y}-\mathbf{DX}||_F^2 \\ & + c_X||\mathbf{X}-\mathbf{X}^{[n-1]}||_F^2 - ||\mathbf{DX}-\mathbf{DX}^{[n-1]}||_F^2 \\ =\ & ||\mathbf{Y}-\mathbf{DX}||_F^2 + \pi_{\mathbf{X}}(\mathbf{X},\mathbf{X}^{[n-1]}) \end{aligned} \tag{15}$$

where $\mathbf{X}^{[n-1]}$ is the coefficient matrix in the previous step, $\pi_{\mathbf{X}}(\mathbf{X},\mathbf{X}^{[n-1]}) := c_X||\mathbf{X}-\mathbf{X}^{[n-1]}||_F^2 - ||\mathbf{DX}-\mathbf{DX}^{[n-1]}||_F^2$ and $c_X > ||\mathbf{D}^T\mathbf{D}||$ is a constant, where $||.||$ is defined as the spectral norm [31]. This type of majorization has already been used for sparse approximation with vector valued coefficients [26], [32], [33]. $\Phi(\mathbf{D},\mathbf{X})$ in (6) has two terms, $||\mathbf{Y}-\mathbf{DX}||_F^2$ and $\lambda J_{p,p}(\mathbf{X})$. Therefore a function majorizing $\Phi(\mathbf{D},\mathbf{X})$ is,

$$\Phi(\mathbf{D},\mathbf{X}) \leq \Phi(\mathbf{D},\mathbf{X}) + \pi_{\mathbf{X}}(\mathbf{X},\mathbf{X}^{[n-1]}) \tag{16}$$

Let $\mathbf{A} := \frac{1}{c_X}(\mathbf{D}^T\mathbf{Y} + (c_X\mathbf{I} - \mathbf{D}^T\mathbf{D})\mathbf{X}^{[n-1]})$. It can be shown that the optimum of the surrogate objective (16), where $p = 1$, is found by shrinking elements in $\mathbf{A}$ [26], [34], that is,

$$\{\mathbf{X}^{[n]}\}_{i,j} = \mathcal{S}_\lambda(\mathbf{A}) = \begin{cases} a_{i,j} - \lambda/2\ sign(a_{i,j}) & \lambda/2 < |a_{i,j}| \\ 0 & otherwise. \end{cases} \tag{17}$$

The matrix $\mathbf{A}$ is the modified *Landweber update* [35], (which is a gradient descent update) of the matrix valued coefficients. This iterative update continues until $\mathbf{X}^{[\mathbf{n}]}$ converges to the optimum solution. The pseudocode for this coefficient update is presented in Algorithm 1. The operator $\mathcal{S}_\lambda$ is the shrinkage operator defined in (17).

C. Dictionary Update

In the second step of the alternating minimization, we minimize the objective function with respect to $\mathbf{D}$ keeping $\mathbf{X}$ fixed. This constrained minimization problem can be solved using several methods. Among these, fixed-point iteration and iterative gradient projection methods have been suggested for the dictionary updates in [23], [11]. In this paper we derive a majorization method for the dictionary update.

The quadratic part of the objective function in (6) has a bounded curvature when minimizing over $\mathbf{D}$. So again using the Taylor series, the majorizing function is as follows,

$$\begin{aligned}||\mathbf{Y}-\mathbf{DX}||_F^2 \le\ & ||\mathbf{Y}-\mathbf{DX}||_F^2 \\ & + c_D||\mathbf{D}-\mathbf{D}^{[n-1]}||_F^2 - ||\mathbf{DX}-\mathbf{D}^{[n-1]}\mathbf{X}||_F^2 \\ =\ & ||\mathbf{Y}-\mathbf{DX}||_F^2 + \pi_{\mathbf{D}}(\mathbf{D},\mathbf{D}^{[n-1]})\end{aligned} \tag{18}$$

where $\mathbf{D}^{[n-1]}$ is the dictionary found in the previous step, $\pi_{\mathbf{D}}(\mathbf{D},\mathbf{D}^{[n-1]}) := c_D||\mathbf{D}-\mathbf{D}^{[n-1]}||_F^2 - ||\mathbf{DX}-\mathbf{D}^{[n-1]}\mathbf{X}||_F^2$ and $c_D > ||\mathbf{X}^T\mathbf{X}||$ is a constant. When $\mathbf{X}$ changes in the sparse approximation step, this spectral norm needs to be re-calculated. We know that the spectral norm of a Hermitian matrix is its largest eigenvalue and various efficient methods have been presented to calculate it [36].

This majorizing function can be used with different constraints. In the following two subsections we derive the optimum of (18) under bounded Frobenius and column-norm constraints.

1) Constrained Frobenius-Norm Dictionaries: An advantage of using a constraint on the Frobenius-norm of the dictionary is that the learnt dictionary can have columns with different norms. Such dictionaries can then be used in the weighted-pursuit framework [37], where atoms with large norms have more chance to appear in the approximations. It has been shown that the average performance of sparse approximation increases when the weights are chosen correctly for the class of signals under study [37].

In the dictionary update step, with the help of a Lagrangian multiplier γ, we turn (6) into an unconstrained optimization problem,

$$\min_{\mathbf{D}} \phi_\gamma(\mathbf{D},\mathbf{X}), \tag{19}$$

where $\phi_\gamma(\mathbf{D},\mathbf{X})$, for $p=1$, is now defined as,

$$\phi_\gamma(\mathbf{D},\mathbf{X}) = ||\mathbf{Y}-\mathbf{DX}||_F^2 + \lambda J_{1,1}(\mathbf{X}) + \gamma(||\mathbf{D}||_F^2 - c_F). \tag{20}$$

Fixing $\mathbf{X}$, the solution to this minimization problem is a global minimum if the solution satisfies the K.K.T conditions [38, Theorem 28.1]. As the admissible set is convex, any minimum of $\Phi_\gamma(\mathbf{D},\mathbf{X})$ is an optimal solution if $\gamma(||\mathbf{D}||_F^2 - c_F) = 0$. Therefore if $||\mathbf{D}||_F^2 \neq c_F$, γ must be zero.

The majorizing function is generated by adding $\pi_{\mathbf{D}}$ to the objective function,

$$\psi_\gamma(\mathbf{D},\mathbf{D}^{[n-1]}) = \phi_\gamma(\mathbf{D},\mathbf{X}) + \pi_{\mathbf{D}}(\mathbf{D},\mathbf{D}^{[n-1]}). \tag{21}$$

$\mathbf{X}$ has here been omitted from the list of parameters because it is assumed fixed in the dictionary update step. The optimum of this function is at a point with zero gradient,

$$\begin{aligned}\frac{d}{d\mathbf{D}}\psi_\gamma(\mathbf{D},\mathbf{D}^{[n-1]}) = -2\mathbf{XY}^T + 2\mathbf{XX}^T\mathbf{D}^{[n-1]^T} + 2c_D\mathbf{D}^T \\ - 2c_D\mathbf{D}^{[n-1]^T} + 2\gamma\mathbf{D}^T = \mathbf{0}\end{aligned}$$

Algorithm 2 : $\mathcal{DU}(\mathbf{X}_{t+1},\mathbf{D}_t)$

```
initialization: $c_D > \|\mathbf{X}_{t+1}^T\mathbf{X}_{t+1}\|$ , $\mathbf{D}^{[0]} = \mathbf{D}_t$
for $n = 1$ to $K_D$ do
    $\mathbf{B} = \frac{1}{c_D}(\mathbf{YX}_{t+1}^T + \mathbf{D}^{[n-1]}(c_D\mathbf{I} - \mathbf{X}_{t+1}\mathbf{X}_{t+1}^T))$
    $\mathbf{D}^{[n]} = \mathcal{P}(\mathbf{B})$
end for
output: $\mathbf{D}_{t+1} = \mathbf{D}^{[K_D]}$
```

By solving the above equation we find the optimal dictionary,

$$\mathbf{D}_\gamma^* = \frac{c_D}{\gamma + c_D}\mathbf{B} \tag{22}$$

where $\mathbf{B}$ is defined as

$$\mathbf{B} := \frac{1}{c_D}(\mathbf{YX}^T + \mathbf{D}^{[n-1]}(c_D\mathbf{I} - \mathbf{XX}^T)). \tag{23}$$

$\mathbf{B}$ has again the same role as the *Landweber update*. To satisfy the K.K.T. conditions, a non-negative γ has to be found such that $\gamma(||\mathbf{D}^{[n]}||_F^2 - c_F) = 0$. If $\mathbf{D}_{\mathbf{0}}^* = \mathbf{B}$ is admissible, we can update the dictionary $\mathbf{D}^{[n]} = \mathbf{B}$. Otherwise we scale $\mathbf{B}$ to have Frobenius-norm equal to $c_F^{1/2}$.

$$\mathbf{D}^{[n]} = \mathcal{P}_{c_F}^F(\mathbf{B}) = \begin{cases}\mathbf{B} & ||\mathbf{B}||_F \le c_F^{1/2} \\ \frac{c_F^{1/2}}{||\mathbf{B}||_F}\mathbf{B} & otherwise\end{cases} \tag{24}$$

The pseudocode for this dictionary update is presented in Algorithm 2. Here $\mathcal{P}$ is the operator $\mathcal{P}_{c_F}^F$ presented in (24). In the following, we show that the dictionary updates, subject to the constraints on the column-norms or the joint sparsity (see below) of the dictionaries, have similar algorithms, but with the different operators for $\mathcal{P}$.

If we use an equality in the definition of (7), i.e. we demand a *fixed* Frobenius-norm, γ can become negative. In this case the decision criteria of (24) becomes an equality ($||\mathbf{B}||_F = c_F^{1/2}$).

2) Constrained Column-Norm Dictionaries: Another often used admissible set in dictionary learning is the set of *fixed* or unit column norm matrices. Instead a bound on the column norms of the dictionary can be used to get a convex admissible set. To make (6) an un-constrained optimization problem we need N Lagrangian multipliers (equal to the number of constraints),

$$\min_{\mathbf{D}} \phi_{\mathbf{\Gamma}}(\mathbf{D},\mathbf{X}), \tag{25}$$

where $\phi_{\mathbf{\Gamma}}(\mathbf{D},\mathbf{X})$, for $p=1$, is now defined as,

$$\phi_{\mathbf{\Gamma}}(\mathbf{D},\mathbf{X}) = ||\mathbf{Y}-\mathbf{DX}||_F^2 + \lambda J_{1,1}(\mathbf{X}) + \sum_{i=1}^{N}\gamma_i(\mathbf{d}_i^T\mathbf{d}_i - c_C) \tag{26}$$

With this formulation, the K.K.T conditions are,

$$\forall i: 1 \le i \le N, \qquad \gamma_i(\mathbf{d}_i^T\mathbf{d}_i - c_C) = 0\ . \tag{27}$$

This means that for each i when $\mathbf{d}_i^T\mathbf{d}_i$ is not equal to c_C, γ_i should be zero. (25) can be rewritten as

$$\phi_{\mathbf{\Gamma}}(\mathbf{D},\mathbf{X}) = ||\mathbf{Y}-\mathbf{DX}||_F^2 + \lambda J_{1,1}(\mathbf{X}) + tr\{\mathbf{\Gamma}(\mathbf{D}^T\mathbf{D} - c_C\mathbf{I})\}, \tag{28}$$

where Γ is a diagonal matrix with the γ_i as the i^{th} diagonal element. By adding $\pi_{\mathbf{D}}$, we get the majorizing function,

$$\psi_{\mathbf{\Gamma}}(\mathbf{D},\mathbf{D}^{[n-1]}) = \phi_{\mathbf{\Gamma}}(\mathbf{D},\mathbf{X}) + \pi_{\mathbf{D}}(\mathbf{D},\mathbf{D}^{[n-1]}). \tag{29}$$

The gradient is again set to zero and the optimum solution is found to be,

$$\mathbf{D}^*_{\Gamma} = \mathbf{B}(\frac{1}{c_D}\mathbf{\Gamma} + \mathbf{I})^{-1}, \tag{30}$$

where $\mathbf{B}$ has the same definition as introduced in (23). All γ_i are non-negative and $(\frac{1}{c_D}\mathbf{\Gamma} + \mathbf{I})$ is an (invertible) diagonal matrix. In equation (30), by changing γ_i, we multiply the corresponding column of $\mathbf{B}$ by a scalar. We start by setting all $\gamma_i = 0$. For any columns of $\mathbf{D}^*_{\mathbf{0}} = \mathbf{B}$ for which the norm is more than $c_C^{1/2}$, we find the smallest value of γ_i which scales down that column to have the largest acceptable norm ($c_C^{1/2}$).

$$\begin{aligned} \mathbf{D}^{[n]} &= \mathcal{P}^C_{c_C}(\mathbf{B}) = \{\mathbf{b}^{[n]}_j\}_{1\le j\le N} \\ \mathbf{d}^{[n]}_j &= \begin{cases} \mathbf{b}_j & ||\mathbf{b}_j||_2 \le c_C^{1/2} \\ \frac{c_C^{1/2}}{||\mathbf{b}_j||_2}\mathbf{b}_j & otherwise, \end{cases} \end{aligned} \tag{31}$$

where $\mathbf{d}_j$ and $\mathbf{b}_j$ are the j^{th} columns of $\mathbf{D}$ and $\mathbf{B}$ respectively.

Alternatively, we can use a *fixed* column-norm constraint ($\mathscr{D} = \{\mathbf{D}_{d\times N} : ||\mathbf{d}_i||_2 = c_C^{1/2}\}$). Here the algorithm may find a Γ in which some of the γ_i are negative. The dictionary update can then be found by a similar operator as (31) but with equality in the decision criteria ($||\mathbf{b}_j||_2 = c_C^{1/2}$) or simply by

$$\mathbf{d}^{[n]}_j = \frac{c_C^{1/2}}{||\mathbf{b}_j||_2}\mathbf{b}_j. \tag{32}$$

When the norm of any columns of $\mathbf{B}$ is zero, we have some ambiguity in the update formula. In this case we can shrink the size of the dictionary by deleting this atom or keep the size fixed by introducing a random atom to the dictionary. In practice we have not encountered such an ambiguity.

D. Jointly Sparse Dictionaries

The majorization approach to dictionary learning is extremely flexible. To demonstrate this, we introduce an additional constraint that encourages dictionary size reduction. In some applications there is a benefit in using a smaller dictionary. One of these benefits could be in coding, where the coding cost increases when the size of the dictionary grows. To shrink the dictionary size during learning, we introduce the following additional constraint on the number of atoms in the dictionary.

$$\min_{\mathbf{X},\mathbf{D}\in\mathscr{D}} \phi_{\theta,0,\infty}(\mathbf{D},\mathbf{X}) \ ;$$

$$\phi_{\theta,0,\infty}(\mathbf{D},\mathbf{X}) = ||\mathbf{Y}-\mathbf{D}\mathbf{X}||^2_F + \lambda J_{1,1}(\mathbf{X}) + \theta|| \max_i |\{\mathbf{D}\}_{i,j}| ||_0$$

where $||.||_0$ is an operator that counts the number of non-zero elements. Because $\phi_{\theta,0,\infty}$ is non-convex and non-continuous, we replace the objective function with a relaxed version as follows,

$$\min_{\mathbf{X},\mathbf{D}\in\mathscr{D}} \phi_{\theta,1,q}(\mathbf{D},\mathbf{X}) \ ;$$

$$\phi_{\theta,1,q}(\mathbf{D},\mathbf{X}) = ||\mathbf{Y}-\mathbf{D}\mathbf{X}||^2_F + \lambda J_{1,1}(\mathbf{X}) + \theta J_{1,q}(\mathbf{D}^T) \tag{33}$$

This objective is convex when $\mathbf{X}$ is fixed. For fixed $\mathbf{X}$, to minimize over $\mathbf{D}$, the joint sparsity penalty is again decoupled by adding $\pi_{\mathbf{D}}$, (defined above), to the objective function

$$\psi_{\theta,1,q}(\mathbf{D},\mathbf{D}^{[n-1]}) = \phi_{\theta,1,q}(\mathbf{D},\mathbf{X}) + \pi_{\mathbf{D}}(\mathbf{D},\mathbf{D}^{[n-1]}). \tag{34}$$

By separating the terms depending on $\mathbf{D}$, the surrogate cost can be written as,

$$\psi_{\theta,1,q}(\mathbf{D},\mathbf{D}^{[n-1]}) \propto c_s tr\{\mathbf{D}\mathbf{D}^T - 2\mathbf{B}\mathbf{D}^T\} + J_{1,q}(\mathbf{D}^T) \tag{35}$$

where $\mathbf{B}$ is defined in (23). The dictionary constraint is again introduced into the objective function using Lagrangian multiplier(s). Let $\mathbf{d}_j$ and $\mathbf{b}_j$ be the j^{th} columns of $\mathbf{D}$ and $\mathbf{B}$ respectively. The objective function, using the bounded column-norm (8), can be written as,

$$\begin{aligned} \psi_{\theta,1,q}(\mathbf{D},\mathbf{D}^{[n-1]}) &\propto \sum_j (tr\{\tau_j^2\mathbf{d}_j\mathbf{d}_j^T - 2\mathbf{b}_j\mathbf{d}_j^T\} + \tfrac{\theta}{c_D}||\mathbf{d}_j||_q) \\ &= \sum_j (\tau_j^2\mathbf{d}_j^T\mathbf{d}_j - 2\mathbf{d}_j^T\mathbf{b}_j + \tfrac{\theta}{c_D}||\mathbf{d}_j||_q) \\ &\propto \sum_j ((\tau_j\mathbf{d}_j - \mathbf{b}_j/\tau_j)^2 + \tfrac{\theta}{c_D\tau_j}||\tau_j\mathbf{d}_j||_q) \\ &= \sum_j \psi_q^{\frac{\theta}{c_D\tau_j}}(\tau_j\mathbf{d}_j, \mathbf{b}_j/\tau_j) \end{aligned} \tag{36}$$

where $\psi_q^\alpha(\mathbf{v},\mathbf{w}) = (\mathbf{w}-\mathbf{v})^2 + \alpha||\mathbf{v}||_q$, $\tau_j = (1+\gamma_j/c_D)^{1/2}$ and γ_j are the Lagrangian multipliers. To minimize (36), we can minimize the first term by minimizing ψ_q^α for each $\mathbf{d}_j$ independently. With the help of two lemmas presented in [39], we can find the optimum of ψ_q^α based on $\mathbf{d_j}$ for $q = 1, 2$ and ∞. The minimum of $\psi_q^\alpha(\mathbf{v},\mathbf{w})$ based on $\mathbf{v}$ [39, Lemma 4.1] is,

$$\min_{\mathbf{v}} \psi_q^\alpha(\mathbf{v},\mathbf{w}) = \mathbf{w} - \mathscr{P}_\alpha^{q'}(\mathbf{w}) \tag{37}$$

where $\mathscr{P}_\alpha^{q'}$ is the orthogonal projection onto the dual norm ball with radius $\mathbf{w}$ and the dual norm is defined as $||.||_{q'}$ with $1/q' + 1/q = 1$. This minimization problem can be solve analytically for some q [39, Lemma 4.2]. In this paper we derive the dictionary update formula for $q = 2$. Interested readers can derive the update formulas when $q = 1$ or $q = \infty$ in the same way. We have

$$\begin{aligned} \mathbf{B}^*_\tau &= \{\mathbf{b}^*_j\}_{1\le j\le N} \\ \mathbf{b}^*_j &= \arg\min_{\mathbf{d}_j} \psi_2^{\frac{\theta}{c_s\tau_j}}(\tau_j\mathbf{d}_j, \mathbf{b}_j/\tau_j) \\ &= \begin{cases} \frac{1}{\tau_j^2}(1 - \frac{\theta}{2c_D||\mathbf{b}_j||_2})\ \mathbf{b}_j & \frac{\theta}{2c_D} < ||\mathbf{b}_j||_2 \\ 0 & otherwise\ , \end{cases} \end{aligned} \tag{38}$$

where $\tau = \{\tau_j\}_{1\le j\le N}$. When all γ_j are non-negative, for any inadmissible $\mathbf{b}^*_j$ with $\tau_j = 1$ ($\gamma_j = 0$), one can decrease $||\mathbf{d}^*_j||_2$ to $c_c^{1/2}$ by increasing τ_j to satisfy the K.K.T conditions. Let $\mathcal{S}^J_{\frac{\theta}{c_D}}(\mathbf{B}) := \mathbf{B}^*_{\tau=\mathbf{1}}$ for any $\mathbf{B}$ found by (23). The dictionary update is therefore done by $\mathcal{P}^C_{c_C}\mathcal{S}^J_{\frac{\theta}{c_D}}(\mathbf{B})$.

When we are looking for a bounded Frobenius-norm dictionary, the dictionary update could be derived, using a similar approach, by $\mathcal{P}^F_{c_F}\mathcal{S}^J_{\frac{\theta}{c_D}}(\mathbf{B})$.

Algorithm 3 : $\mathcal{DL}(\mathbf{X}_0, \mathbf{D}_0)$

```
for $t = 1$ to $T$ do
    $\mathbf{X}_{t+1} = \mathcal{SA}(\mathbf{X}_t, \mathbf{D}_t)$
    $\mathbf{D}_{t+1} = \mathcal{DU}(\mathbf{X}_{t+1}, \mathbf{D}_t)$
end for
output: $\mathbf{D}_T$
```

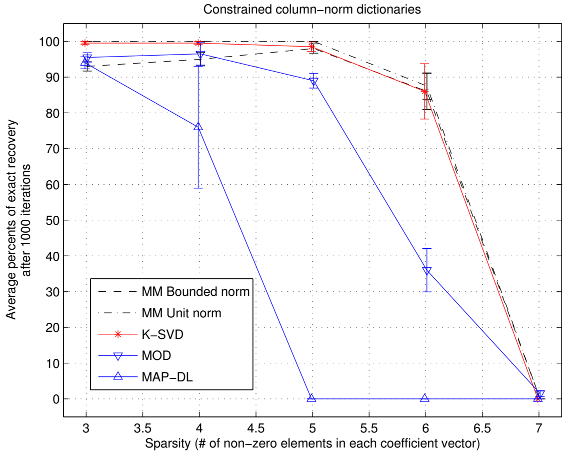

Fig. 1. A comparison of the dictionary recovery success rates using different dictionary learning methods under a column-norm constraint.

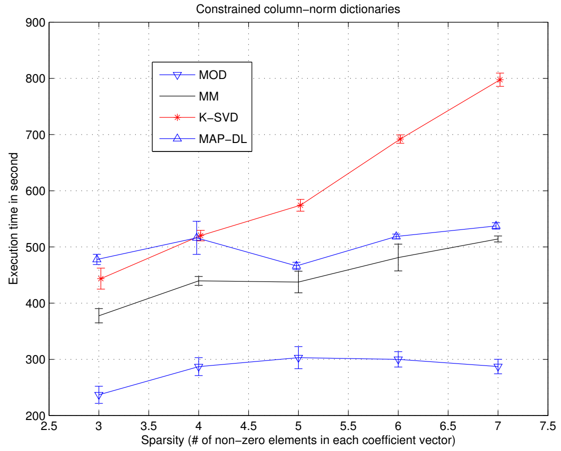

Fig. 2. A comparison of the computation costs of the dictionary learning methods under a column-norm constraint.

E. Generalized block relaxation method for dictionary learning

In the previous subsections we presented a block relaxation method to optimize $\mathbf{X}$ and $\mathbf{D}$ iteratively. In each step, we used an iterative method to find the optimum solution based on one variable while keeping the other variable fixed. The pseudocode for dictionary learning in this framework is presented in Algorithm 3.

Because the joint objective function does not have a fixed bounded curvature, we could not use the majorization method for both parameters jointly. On the other hand, this alternating optimization decreases the rate of convergence as it often oscillates around the optimal path. Instead of fully optimizing with respect to a single parameter in each step, the generalized block relaxation method updates each variable at a time and reduces the objective function, using for example a cyclic selection or any other periodic selection of the parameters. A simple way to choose which parameter to update is to calculate the update based on each parameter and then choose the parameter that decrease the objective function the most. A drawback of this type of parameter selection is that it doubles the computational cost. Another technique is to alternatively update each parameter. For dictionary learning, we found that using more coefficient updates than dictionary updates is in general more beneficial. So one can use p updates of $\mathbf{X}$ followed by q updates of $\mathbf{D}$ when $p \geq q$.

A more complete explanation and a basic convergence proof for the generalized block relaxed dictionary learning algorithm are provided in Appendix B. It is easy to show that the block relaxation method is a special case of the generalized block relaxation method. Therefore convergence of the block relaxation method (alternating minimization) for the dictionary learning follows as a corollary of this result.

IV. SIMULATIONS

We evaluate the proposed method with synthetic and real data. Using synthetic data with random dictionaries helps us to examine the ability of the proposed methods to recover dictionaries exactly (to within an acceptable squared error). We generated the synthetic data and dictionaries as proposed in [23] and [15]. To evaluate the performance on real data, we chose audio signals, which have been shown to have some sparse structure. We then used the learnt dictionary for audio coding and show some improvements in Rate-Distortion performance compared to coding with classical dictionaries.

A. Synthetic Data

A 20×40 matrix $\mathbf{D}$ was generated by normalizing a matrix with i.i.d. uniform random entries. The number of non-zero elements in each of the coefficient vectors was selected between 3 and 7. The locations of the non-zero coefficients were selected uniformly at random. We generated 1280 training samples where the absolute values of the non-zero coefficients were selected uniformly between 0.2 and 1. In the setting for exact dictionary recovery [15], [23] and under a mild condition, the constrained column-norm dictionary and the K-sparse signals are the global solutions of the dictionary learning problem based on exact sparse representations and the ℓ_1 based exact sparse representation problems, respectively (see for example [19]). The proposed algorithm as well as the other dictionary learning algorithms discussed, are proposed for sparse *approximations*, that is, they allow approximation error when calculating the sparse coefficients. To adapt the algorithm to this problem, we assumed that the sparse approximation finds the correct support in each step. Once the support has been identified, we find the best approximation by projecting onto the selected sub-space. This is called debiasing.

We here compare the majorization based dictionary learning

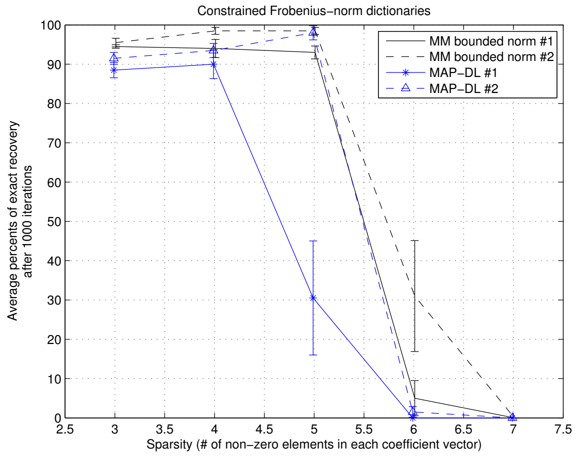

Fig. 3. A comparison of the dictionary recovery success rates using MM and MAP dictionary learning methods under a Frobenius norm constraint: 1: Desired dictionary had fixed Frobenius-norm. 2: Desired dictionary had fixed column-norms.

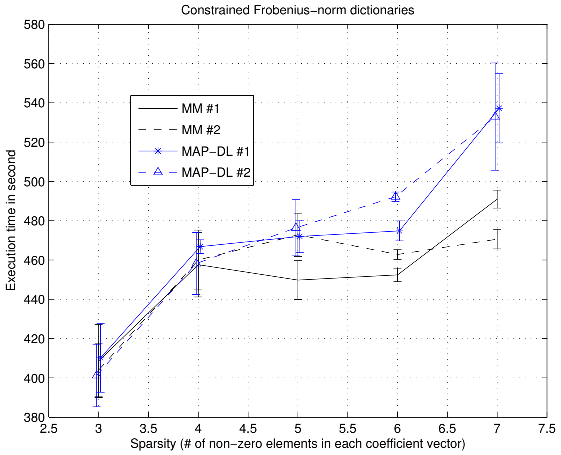

Fig. 4. A comparison of the computation costs of the dictionary learning methods under a Frobenius norm constraint.

algorithm to MOD, K-SVD and MAP-DL. The stopping criteria for IT was the distance between two consecutive iterations ($\delta = 3\times10^{-4}$) and λ was set to 0.4. The termination conditions for the iterative dictionary learning methods (majorization method for dictionary learning (MM-DL) and MAP-DL) was set to ($||\mathbf{D}^{[n]} - \mathbf{D}^{[n-1]}||_F \leq 10^{-7}$).

We started from a normalized random $\mathbf{D}$ and used 1000 iterations. The learning parameter (γ) in MAP-DL was selected as described in [23] and we down-scaled γ by a factor of 2^{-j} ($j > 1$) when the algorithm was diverging. To allow a fair comparison, we repeated the simulations 5 times. If the squared error between a learnt and true dictionary element was below 0.01, it was classified as correctly identified. The average percentages and standard deviations are shown in Figure 1. It can be seen that in all cases, MM-DL with fixed column-norm and K-SVD recovered nearly the same number of atoms and performed better than the other methods (although, for the signals with less than 6 non-zero coefficients, MM-DL recovered all desired atoms, performance of K-SVD was very close to it). The debiasing process creates some ambiguities in dictionary learning when using the bounded-norm constraints as they reduce the effect of the coefficient magnitudes in the sparsity measure. Therefore, we observe atoms which do not have a boundary norm (here, unit norm), even after 1000 iterations. In this case, we get better results using a fixed column-norm admissible set which resolves this ambiguity. The MAP-DL algorithm did not perform well in this simulation. We guess the reason for this is slow convergence of the approach and the use of more iterations might improve the performance.

In Fig.2 we compare the computation time of the algorithms for the above simulations. Simulations ran on the Intel Xeon 2.66 GHz dual-core processor machine and both cores were used by Matlab. In this graph the total execution time of the algorithms (sparse approximations plus dictionary updates for 1000 iterations) is shown. MOD was fastest followed by our MM-DL.

We have a larger admissible set when fixing the Frobenius-norm of the dictionary, which makes the problem of exact recovery more complicated and we expect to observe worse performance in terms of exact atom recovery. To test this, we started with a normalized random dictionary, normalized either to have fixed Frobenius-norm or fixed column-norm. The simulations were repeated for 5 trials and the averages and standard deviations of the atom recovery are shown in Fig. 3. In these simulations MM-DL performed slightly better than MAP-DL. The other observation in this figure is that when the desired dictionaries have equal column-norms, performance of the algorithms increase but do not reach the performance observed when using the more restricted (and appropriate) admissible set. Computation times of the algorithms, on the machine described formerly, are shown in Fig.4.

In the next experiment we assume that the desired dictionary size is unknown but bounded. We generated the data as in the previous experiments but the simulations were started with four times overcomplete dictionaries (two times larger than the desired dictionary size). The dictionary updates were based on the joint sparsity objective function (33) (with $\theta = 0.05$, $p = 1$ and $q = 2$). The average percentage of exact atom recovery for 5 trials are shown in Fig. 5 and 6. We plotted the percentage of the exact recovery of the original atoms, regardless of the learnt dictionary size. In the lower plot, we show the size of dictionary after 1000 iterations. With this θ we identified the size correctly but for less sparse signals (higher k) we got less accurate results. The overall performance of the algorithm is determined by the correct choice of θ. By increasing θ we find smaller dictionaries and vice versa.

B. *Dictionary Learning for Sparse Audio Coding*

In this section we demonstrate the performance of the proposed dictionary learning method on audio signals and thus show that our method is applicable to large dictionary learning problems. An audio sample of more than 8 hours was recorded from BBC radio 3, which plays mostly classical music.

In the first experiment we used bounded column-norm and bounded Frobenius-norm dictionary admissible sets. The audio

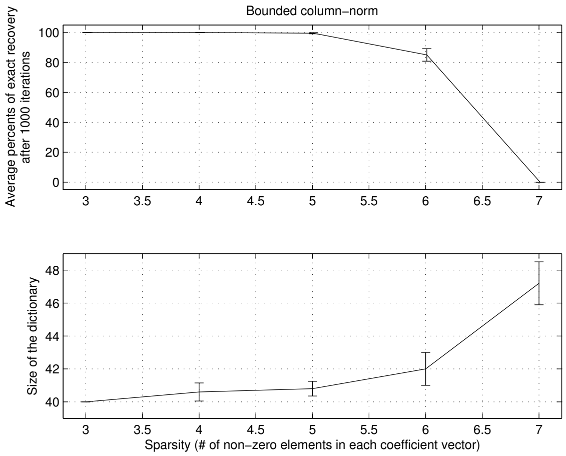

Fig. 5. Dictionary recovery success rates under a column-norm constraint and joint sparsity penalty.

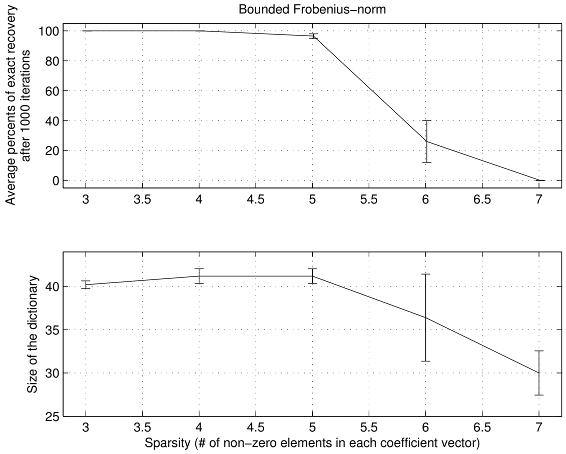

Fig. 6. Dictionary recovery success rates under a Frobenius norm constraint and joint sparsity penalty.

sample was summed to mono and down-sampled by a factor of 4. From this 12kHz audio signal, we randomly took 4096 blocks of 256 samples each. The set of dictionaries with the column-norms bounded by c_C is a subset of the set of bounded Frobenius-norm dictionaries, when $c_F = Nc_C$. We chose dictionary admissible sets with column-norms and Frobenius-norms bounded by $c_C = 1$ and $c_F = N$ respectively. We initialized the dictionary with a 2 times overcomplete random dictionary and used 1000 iterations. The objective function against iteration, for two different values of λ, are shown in Fig. 7. This figure shows that the optimal bounded Frobenius-norm dictionaries are better solutions for the objective functions.

As a second experiment, we looked at an audio coding example. We used the proposed method with the bounded Frobenius-norm constraint to learn a dictionary based on a training set of 8192 blocks, each 1024 samples long. In this experiment we want to learn the dictionary for a larger block length than the previous experiment. The convergence of the traditional block relaxation method for a problem with this size is very slow. Therefore we run the simulations with the generalized block relaxation method and a joint sparsity constraint on the dictionary to encourage shrinkage of the dictionary. This shrinkage makes the algorithm faster in later iterations. Even though the recorded audio had 48k samples per second, the audio had a maximum frequency of 16kHz. Therefore we downsampled the original audio by a factor of 3/2 without any degradation in the audio fidelity. It has been shown that audio can be modeled reasonably well using tonal, transient and noisy residual components [40]. We chose a 2 times overcomplete sinusoid dictionary (frequency oversampled DCT) as the initialization point and ran the simulations with different lambda values for 5000 iterations of alternative optimization of (41), which took approximately 8 hours for each λ, running on the machine mentioned in the previous

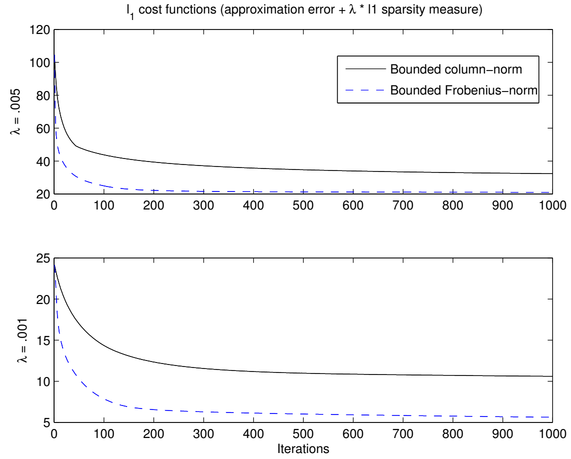

Fig. 7. ℓ_1 cost functions for two different Lagrangian multipliers (λ) .005 (top) and .001 (bottom).

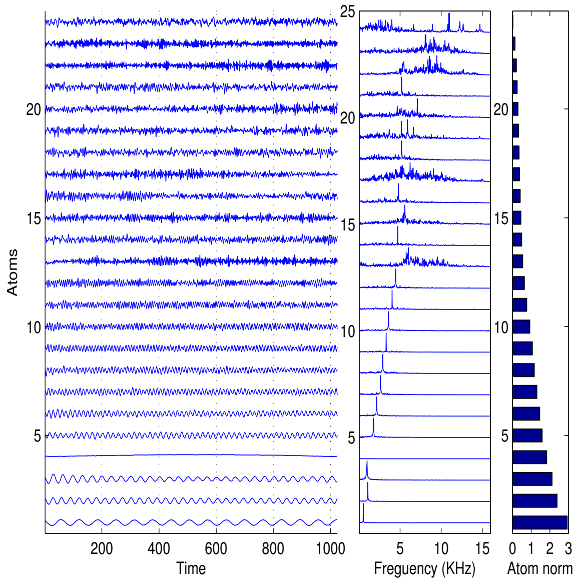

Fig. 8. A selection of learnt atoms in time (left) and frequency (middle) domain. Their norms are shown in the right panel.

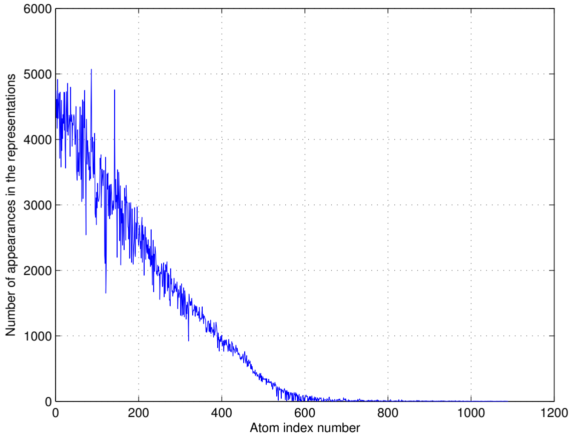

Fig. 9. Number of appearances of the learnt atoms in the representations of the training samples (of size 8192).

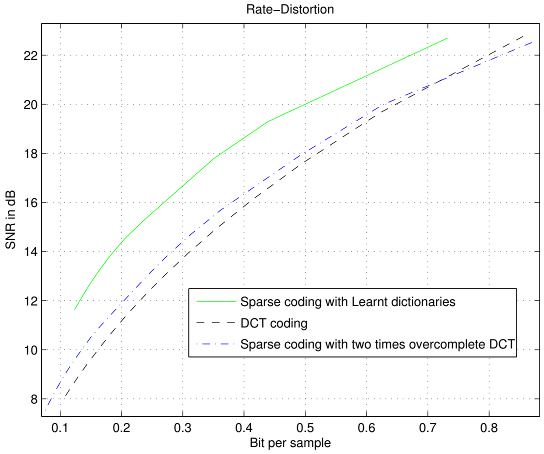

Fig. 10. Estimated Rate-Distortion for the audio coding example using the learnt dictionary, the shrunk 2 times overcomplete DCT dictionary and the DCT.

subsection.

A subset of the learnt atoms ($\lambda = .01$, $\theta = .01$), which is selected by uniformly sampling the atom indices, is shown in Fig. 8. These atoms are shown in the time and frequency domain in the left and middle windows respectively. The norms of the selected atoms are shown in the right window. The number of appearances of each atom, which are sorted based on their ℓ_2 norms, are shown in Fig. 9. To design an efficient encoder we only used atoms that were used frequently in the representations. Therefore we were able to further shrink the dictionary size. In this test we chose a threshold of 40 appearances (out of 8192) as the selection criteria. This dictionary was used to find the sparse approximations of 4096 different random blocks, each of 1024 samples, from the same data set. We then encoded the location (significant bit map) and magnitude of the non-zero coefficients separately. In this paper we used a uniform scalar quantizer with a double zero bin size to code the magnitude. We estimated the entropy of the coefficients to approximate the required coding cost. To encode the significant bit map, we assumed an i.i.d. distribution for the location of the non-zero atoms. The same coding strategy was used to code sparse approximations with a two times frequency overcomplete DCT (the initial dictionary used for learning) followed by shrinking based on the number of appearances. For reference we calculated the rate-distortation of the DCT coefficient encoding of the same data, using the same method of significant bitmap and non-zero coefficients coding. The performance is compared in Fig. 10. In the sparse coding methods, the convex hulls of the rate-distortion performances calculated with different dictionaries, each optimized and shrunk for different bit-rate, are shown in this figure. Using the learnt dictionaries for sparse approximation is superior to using the DCT or overcomplete DCT for the range of bit-rates shown.

It would be nice to compare these real data experiments with K-SVD, which is shown to perform well in dictionary learning for medium size problems. However, we found K-SVD to be too slow on problems of this size. For example, one sparse approximations of the signals, using a fast implementation of OMP [41], and one dictionary update approximately took 10 hours and this has to be repeated for a reasonable number of iterations, e.g. 1000 iterations!

V. Conclusions

We have presented a new algorithm for dictionary learning and have shown its advantages with different experiments and for different data sets. The proposed method is very flexible in using different constraints on the dictionaries. Because the problem of dictionary learning is considered in a more general form (bounded norm for dictionaries), better results were possible.

While some of the other methods are based on atom-wise dictionary update (K-SVD, MAP-DL with unit column-norm *a priori* information), the proposed method updates the whole dictionary at once. Although the computational complexity of each iteration of the given algorithm is roughly cubic, we found that the algorithm is much faster for large scale problems than, for example, K-SVD (which has a higher order of complexity).

The given method solves the dictionary learning problem in a unified framework. This unified framework provides extra flexibility to update the coefficients and the dictionary in a more efficient way. Furthermore, we showed the convergence of the method to a set of fixed points in this framework.

Finally we have shown that the constrained Frobenius-norm can increase the performance of dictionary learning by increasing the possible solution set. Audio coding with the learnt dictionary showed a superior rate-distortion performance over traditional orthogonal transform coding and overcomplete sparse coding with an oversampled DCT.

Appendix A
Matrix Form of the Majorizing Function

We can use the Taylor series to majorize the quadratic term of the objective function which has a bounded curvature. The

Taylor series in matrix form [42, Appendix D 1.7] is given by,

$$f(\mathbf{U}) = f(\mathbf{V}) + \overset{\rightarrow\mathbf{U}-\mathbf{V}}{df}(\mathbf{V}) + \frac{1}{2!}\overset{\rightarrow\mathbf{U}-\mathbf{V}}{df^2}(\mathbf{V}) + o(||\mathbf{U}||^3) \qquad (39)$$

where $\overset{\rightarrow\mathbf{U}-\mathbf{V}}{df}(\mathbf{V})$ and $\frac{1}{2!}\overset{\rightarrow\mathbf{U}-\mathbf{V}}{df^2}(\mathbf{V})$ are the directional first and second derivatives of f at $\mathbf{V}$ in the $\mathbf{U}-\mathbf{V}$ direction. The directional derivatives are defined by,

$$\overset{\rightarrow\mathbf{Y}}{df}(\mathbf{X}) = \{\frac{d}{dt}f(\mathbf{X}+t\mathbf{Y})\}_{t=0}, \quad \overset{\rightarrow\mathbf{Y}}{df^2}(\mathbf{X}) = \overset{\rightarrow\mathbf{Y}}{df}\,(\overset{\rightarrow\mathbf{Y}}{df}\,(\mathbf{X})).$$

For a bounded curvature objective function we have,

$$f(\mathbf{U}) \leq f(\mathbf{V}) + \overset{\rightarrow\mathbf{U}-\mathbf{V}}{df}(\mathbf{V}) + \frac{1}{2}tr\{(\mathbf{U}-\mathbf{V})^T\Pi(\mathbf{U}-\mathbf{V})\}, \quad (40)$$

where $\Upsilon = \Pi - \overset{\rightarrow\mathbf{U}-\mathbf{V}}{df^2}(\mathbf{V})$ is positive definite ($\Upsilon \succ 0$).

Appendix B
Convergence study of the algorithm

In the first step of analyzing an iterative algorithm, we need to show the boundedness of the solutions (or the stability of the algorithm). The stability of the algorithms, in which a positive objective is reduced in each iteration, is guaranteed using Lyapunov's second theorem. For example the stability of the MAP-DL is guaranteed when a *suitable* step size is chosen (to the authors knowledge, no analytical study has been done on how to choose this step size). The convergence of the alternating (gradient) projection based methods essentially depends on the admissible sets (and the gradient step size). In the dictionary learning problem with the admissible sets given by [13] [11], the convergence of the algorithm is not guaranteed. In K-SVD, one needs to find the sparse approximations based on the ℓ_0 sparsity measure for which no efficient algorithm exists so that the stability analysis is challenging. In practice we observed that in MOD and K-SVD, when the solution sequence enters a neighborhood of a local minimum, the objective increases in some iterations. Therefore, it does not converge monotonically to the solution.

The next step is to show the convergence of the algorithm to a fixed point or a set of fixed points. The authors in [23] referred to the convergence of the gradient flow method to show the convergence of the MAP-DL. Although this statement is completely correct, it requires the use of an arbitrary small step size which is practically impossible.

The stability of dictionary learning based on the majorization method has already been proven by the fact that we reduce the objective in each step. Here, we show the convergence to a set of fixed points. Our dictionary learning framework can be viewed as a generalized block-relaxed minimization scheme applied to an augmented objective function. Specifically, we combine two majorizing objectives, (15) and (18),

$$\psi(\mathbf{D},\mathbf{X},\mathbf{D}^{\ddagger},\mathbf{X}^{\ddagger}) = \phi(\mathbf{D},\mathbf{X}) + c_D||\mathbf{D}-\mathbf{D}^{\ddagger}||_F^2 + c_X||\mathbf{X}-\mathbf{X}^{\ddagger}||_F^2 - ||\mathbf{D}\mathbf{X}-\mathbf{D}^{\ddagger}\mathbf{X}^{\ddagger}||_F^2 \qquad (41)$$

where $\mathbf{X}^{\ddagger}$ and $\mathbf{D}^{\ddagger}$ are two auxiliary parameters corresponding to $\mathbf{X}$ and $\mathbf{D}$ respectively. c_D and c_X have been chosen to be larger than the spectral norms of $\mathbf{X}^{\ddagger T}\mathbf{X}^{\ddagger}$ and $\mathbf{D}^{\ddagger T}\mathbf{D}^{\ddagger}$ respectively. This augmented objective function *does not* majorize the joint objective, however when $(\mathbf{D},\mathbf{D}^{\ddagger}|_{\mathbf{D}^{\ddagger}=\mathbf{D}})$ or $(\mathbf{X},\mathbf{X}^{\ddagger}|_{\mathbf{X}^{\ddagger}=\mathbf{X}})$ are fixed, (41) majorizes the original joint objective based on the other pair of parameters. When the optimization method is viewed in the block relaxation framework, the optimum of $\mathbf{X}^{\ddagger}$ or $\mathbf{D}^{\ddagger}$ is easily found by $\mathbf{X}$ or $\mathbf{D}$ respectively. This corresponds to the parameter update in the standard majorization method [29]. Therefore any sequence of updates is acceptable, given each update of $\mathbf{D}$ (or $\mathbf{X}$) is followed by an update based on $\mathbf{D}^{\ddagger}$ (or $\mathbf{X}^{\ddagger}$) respectively.

Such a block-relaxed sequential constrained minimization is not in general guaranteed to converge (see [24] for some counter examples). To study the convergence of our algorithm, we need to do a little more work. In the next subsection, we introduce some theoretical analysis of the generalized block relaxation method. We then analyze the proposed algorithm for dictionary learning, based on the given theoretical analysis.

A. Generalized Block relaxed iterative mappings and their convergence

Let $\eta(\omega):\Omega\rightarrow\mathbb{R}$ be the multi-parameter objective function which we want to minimize. Let Υ be the set of admissible parameters. The parameter ω is defined as the concatenation of the blocks of parameters $\{\omega\in\Upsilon:\ \omega=(\omega_1,\omega_2,...,\omega_p)\ ,\omega_i\in\Omega_i\}$ where $\Omega=\Omega_1\times\Omega_2\times...\times\Omega_p$. In dictionary learning based on block relaxation, $p=2$, $\omega_1=\mathbf{X}$ and $\omega_2=\mathbf{D}$. In generalized block-relaxed dictionary learning, $p=4$ as we have two more auxiliary parameters $\mathbf{X}^{\ddagger}$ and $\mathbf{D}^{\ddagger}$.

We now need to introduce point to set maps,

***Definition* B.1** (Point to set map)**.** Let Υ be an arbitrary set and let Γ be the set of all subsets of Υ. A map $\Delta:\Upsilon\rightarrow\Gamma$ is a point to set map (see for example [43]).

In the block relaxation technique a set of point to set maps $\Delta_i:\Upsilon\rightarrow\Gamma$ are defined as $\Delta_i(\widehat{\omega})=\{\omega\in\Upsilon:\forall j\neq i\ \omega_j=\widehat{\omega}_j\}$ where $\widehat{\omega}=(\widehat{\omega}_1,\widehat{\omega}_2,...,\widehat{\omega}_p)$ is the current value of the parameters. These point to set maps keep all the blocks of parameters fixed apart from the i^{th} block.

By starting from $\omega^{[0]}$, the set of possible solutions Λ in the minimization problem is defined as, $\Lambda=\{\omega\in\Upsilon:\eta(\omega)\leq\eta(\omega^{[0]})\}$. For any $\omega\in\Lambda$ in each block update we minimize the objective for the selected parameters. This gives us the following updating operator:

$$U_i:\Lambda\rightarrow\{u\in\Delta_i(\widehat{\omega}):\eta(u)\leq\eta(t),\forall t\in\Delta_i(\widehat{\omega})\} \qquad (42)$$

In general this updating operator is a point to set map and we can choose an update parameter within the resulting set. In our case, the objective function always has a unique minimizer and the updating operators are point-to-point mappings. To use a set of updating operators, we also need to have an operator selector.

***Definition* B.2** (Operator selector)**.** $s(k):\mathbb{N}\rightarrow\mathscr{P}$ which $\mathscr{P}=\{i:1\leq i\leq p\}$

This operator can choose the updating operator by sequentially selecting (circular) or free steering through the available operators. By using the updating operators defined in (42) and an update selector $s(k)$, we can summarize the (generalized) block relaxed minimization by the following algorithm.

***Algorithm* B.1.** *Let $\omega^{[0]}$ be a given starting point, then $\{\omega^{[k]}\}_{k\in\mathbb{N}}$ is the sequence of updates given by $\omega^{[k+1]} \in U_{s(k)}\{\omega^{[k]}\}$ and stop when $\forall i \in \mathscr{P} : \widehat{\omega} = U_i\{\widehat{\omega}\}$*

When the updating operator is injective, $\omega^{[k+1]} = U_{s(k)}\{\omega^{[k]}\}$, to analyze the sequence generated by Algorithm B.1, we need to introduce some characteristics of the infinite series.

***Definition* B.3** (Asymptotically regularity)**.** A sequence $\{\alpha^{[n]}\}_{n\in\mathbb{N}}$ is asymptotically regular if $||\alpha^{[n+1]} - \alpha^{[n]}|| \to 0$, when $n \to \infty$.

$||\ .\ ||$ is a norm defined in the solution space. An operator is called asymptotically regular when the series generated by the sequential use of that operator is asymptotically regular.

***Definition* B.4** (Essentially periodic)**.** An infinite sequence $\{\alpha^{[n]}\}_{n\in\mathbb{N}}$ drawn from a finite alphabet $\mathscr{P} = \{\mathscr{A}_i : 1 \leq i \leq p\}$ is essentially periodic, with a period $m \in \mathbb{N}, m \geq p$ when $\forall j \in \mathbb{N}, \forall \mathscr{A}_i \in \mathscr{P}, \exists n \in [jm+1, (j+1)m]$ and $\alpha^{[n]} = \mathscr{A}_i$.

The sequence of $\{\omega^{[k]}\}$ of the Algorithm B.1 is asymptotically regular when Δ_i and η satisfy the following hypotheses [44],

***Hypotheses* B.1.** For all $i \in \mathscr{P}$ and $\eta : \Upsilon \to \mathbb{R}$,

- $\forall \omega : \omega \in \Delta_i(\omega)$
- Δ_i is continuous on Υ
- $\forall \omega \in \Upsilon$, η has a unique minimizer over $\Delta_i(\omega)$
- $\exists \omega^{[0]} \in \Upsilon$ such that Λ is a compact subset.

We now study the accumulation points of Algorithm B.1, when the Hypotheses B.1 are satisfied. From basic mathematical analysis, we know that any bounded sequence has at least one accumulation point (Bolzano-Weierstrass Theorem [45, Theorem 4.1]). As Λ is closed, the accumulation points of $\{\omega^{[n]}\}$ are in Λ.

***Theorem* B.1.** *[44, Theorem 15] Let the update selector, $s(k)$, be essentially periodic and Δ_i and η satisfy Hypotheses B.1. Every accumulation point ω^* of $\{\omega^{[n]}\}$, generated by Algorithm B.1, satisfies $\omega^* = U_i\{\omega^*\}$ for any $i \in \mathscr{P}$*

The set of accumulation points T belongs to a level set of η. If η is continuous, T is closed and as Λ is bounded and $T \subseteq \Lambda$, T is bounded. Therefore T is compact.

***Proposition* B.1.** [29, Proposition 10.3.1] If a bounded sequence $\{\omega^{[n]}\}_{n\in\mathbb{N}}$ is asymptotically regular, then its set of accumulation points is connected. If this set is finite, then it reduces to a single point.

In a normed space, the following lemma guarantees that the sequence $\{\omega^{[n]}\}_{n\in\mathbb{N}}$ generated by Algorithm B.1 will stay arbitrarily close to the accumulation points, when $n > N$ for some N.

***Lemma* B.1.** *Let $\{\omega^{[n]}\}_{n\in\mathbb{N}}$ be a bounded asymptotically regular sequence and T be the set of its accumulation points then, $\forall \epsilon > 0, \exists N \in \mathbb{N}, for\ n > N, \exists t \in T, ||\omega^{[n]} - t|| < \epsilon$*

Proof: Let S be an ϵ-neighborhood of T and S_c be its complement in the admissible set. As the admissible set is compact, S_c is also compact. Because S is a neighborhood of T there is no accumulation point t in S_c. If $\{\omega^{[n]}\}$ has infinitely many points in S_c, then it has a converging subsequence and at least one accumulation point in S_c . This contradicts the fact that there is no accumulation point in S_c. Therefore $\exists N : \omega^{[n]} \in S, \forall n > N$. On the other hand ϵ-neighborhood implies that for all $n > N$, $\exists t \in T$: $||\omega^{[n]} - t|| < \epsilon$. ■

In the next subsection we show asymptotic regularity of the generalized block relaxation method for dictionary learning. This is followed by showing the convergence of the proposed method to a set of fixed points.

B. Convergence study of the generalized block-relaxed dictionary learning

In dictionary learning, there are two parameters, coefficient matrix and dictionary. In generalized block-relaxed dictionary learning (41), we have four parameters. We mentioned that the augmented function (41) majorizes (6) only when one pair of parameter blocks ($(\mathbf{D}, \mathbf{D}^{\ddagger}|_{\mathbf{D}^{\ddagger}=\mathbf{D}})$ or $(\mathbf{X}, \mathbf{X}^{\ddagger}|_{\mathbf{X}^{\ddagger}=\mathbf{X}})$) is fixed. Therefore $\Delta_{\mathscr{X}} : \mathscr{X} \in \{\mathbf{D},\ \mathbf{X},\ \mathbf{D}^{\ddagger},\ \mathbf{X}^{\ddagger}\}$ are the point to set maps which fix all parameters but $\mathscr{X}$ (from now on we use this indexing for the point to set maps).

***Proposition* B.2.** The generalized block-relaxed minimization of (41) is asymptotically regular when the updates of $\mathbf{D}$ and $\mathbf{X}$ are followed by updating of $\mathbf{D}^{\ddagger}$ and $\mathbf{X}^{\ddagger}$ respectively.

Proof: To show the asymptotic regularity we show that all the hypotheses in Hypotheses B.1 are satisfied. $\Delta_{\mathscr{X}} : \mathscr{X} \in \{\mathbf{D},\ \mathbf{X},\ \mathbf{D}^{\ddagger},\ \mathbf{X}^{\ddagger}\}$ are self contained, i.e. $\widehat{\mathscr{X}} \in \Delta_{\mathscr{X}}\{\widehat{\mathscr{X}}\}$, and continuous. Therefore they satisfy the first two hypotheses. The minimum of (41) based on each parameter is unique (the sparse approximation minimum is reached using soft shrinkage (17) over $\mathbf{A}$ and the dictionary update is reached by one of the operators introduced in (24), (31) or (38) over $\mathbf{B}$). (41) is strictly convex based on $\mathbf{X}^{\ddagger}$ or $\mathbf{D}^{\ddagger}$ when all other parameters are fixed. Therefore minimization based on $\mathbf{D}^{\ddagger}$ or $\mathbf{X}^{\ddagger}$ has a unique solution. Surrogate objective function (41) is a continuous function. When a mapping is continuous, its epigraph Λ is a closed set [38, Theorem7.1]. As the admissible set is a closed set, the intersection of Λ and this set, which is the possible solution set, is closed. On the other hand there is no infinitely large point in Λ (maximum value of $||\mathbf{D}||_F$ and $J_{1,1}(\mathbf{X})$ are bounded based on the dictionary constraints and $\phi(\mathbf{D}^{[0]}, \mathbf{X}^{[0]})/\lambda$ respectively). In an Euclidean space boundedness and closedness are sufficient for a set to be compact. Therefore the hypothesis is satisfied and the sequence of $(\mathbf{D}, \mathbf{X}, \mathbf{D}^{\ddagger}, \mathbf{X}^{\ddagger})^{[i]} : i \in \mathbb{N}$ is asymptotically regular [44]. ■

Finally we present a Proposition which shows the convergence of the proposed algorithm.

***Proposition* B.3.** Generalized block-relaxed dictionary learning converges to a single fixed point $(\mathbf{D}^*, \mathbf{X}^*)$ or gets arbitrary close to a continuum of accumulation points, where each accumulation point satisfies:

- $\psi(\mathbf{D}^*, \mathbf{X}^*, \mathbf{D}^*, \mathbf{X}^*) \leq \psi(\mathbf{D}^*, \mathbf{X}, \mathbf{D}^*, \mathbf{X}^*) : \forall \mathbf{X}$

- $\psi(\mathbf{D}^*, \mathbf{X}^*, \mathbf{D}^*, \mathbf{X}^*) \leq \psi(\mathbf{D}, \mathbf{X}^*, \mathbf{D}^*, \mathbf{X}^*) : \forall \mathbf{D} \in \mathscr{D}$

Proof: Due to Proposition B.2, the sequence generated by generalized block-relaxed dictionary learning is asymptotically regular. Due to Theorem B.1 and Lemma B.1, the algorithm converges either to a fixed point or gets arbitrary close to a continuum of accumulation points. Because any accumulation point of the algorithm is a fixed point for all $U_i : \forall i \in \mathscr{P}$ [44, Theorem 15], $\mathbf{X}^*$ is the best coefficient matrix using dictionary $\mathbf{D}^*$ and $\mathbf{D}^*$ is the best admissible dictionary, using $\mathbf{X}^*$ as the sparse representation. ■

References

[1] I. Daubechies, *Ten lectures on wavelets*. Society for Industrial and Applied Mathematics, 1992.

[2] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, 1989.

[3] V. Goyal, J. Kovacevic, and J. Kelner, "Quantized frame expansions with erasures," *Applied and Computational Harmonic Analysis*, vol. 10, pp. 203–233, 2001.

[4] http://www.compressedsensing.com.

[5] G. Davis, "Adaptive nonlinear approximations," Ph.D. dissertation, New York University, 1994.

[6] S. Mallat and Z. Zhang, "Matching pursuits with time frequency dictionaries," *IEEE Trans. on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[7] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[8] S. Sardy, A. Bruce, and P. Tseng, "Block coordinate relaxation methods for nonparametric wavelet denoising," *Journal of Computational and Graphical Statistics*, vol. 9, no. 2, pp. 361–379, 2000.

[9] M. Elad and A. Bruckstein, "A generalized uncertainty principle and sparse representations in pairs of bases," *IEEE Trans. Information Theory*, vol. 48, no. 9, p. 25582567, 2002.

[10] R. Gribonval and M. Nielsen, "Sparse representations in unions of bases," *IEEE Trans. Information Theory*, vol. 49, no. 12, pp. 3320 – 3325, 2003.

[11] B. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: a strategy employed by V1?" *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[12] M. Lewicki and T. Sejnowski, "Learning overcomplete representations," *Neural Comp*, vol. 12, no. 2, pp. 337–365, 2000.

[13] K. Engan, S. Aase, and J. Hakon-Husoy, "Method of optimal directions for frame design," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1999.

[14] S. Lesage, R. Gribonval, F. Bimbot, and L. Benaroya, "Learning unions of orthonormal bases with thresholded singular value decomposition," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2005.

[15] M. Aharon, E. Elad, and A. Bruckstein, "K-SVD: an algorithm for designining of overcomplete dictionaries for sparse representation," *IEEE Trans. on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[16] K. Engan, K. Skretting, and J. Husoy, "Family of iterative ls-based dictionary learning algorithms, ILS-DLA, for sparse signal representation," *Digital Signal Processing*, vol. 17, no. 1, pp. 32–49, 2007.

[17] E. Smith and M. S. Lewicki, "Efficient coding of time-relative structure using spikes," *Neural Computation*, vol. 17, no. 1, pp. 19–45, 2005.

[18] M. Plumbley, "Dictionary learning for L1-exact sparse coding," in *International Conference on Independent Component Analysis and Signal Separation, ICA*, 2007.

[19] R. Gribonval and K. Schnass, "Some recovery conditions for basis learning by L1-minimization," in *International Symposium on Communications, Control and Signal Processing, ISCCSP*, 2008.

[20] S. Cotter, B. Rao, K. Engan, and K. Kreutz Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Trans. on Signal Processing*, vol. 53, no. 7, pp. 2477–2488, 2005.

[21] M. Figueiredo, J. Bioucas-Dias, and R. Nowak, "Majorization-minimization algorithms for wavelet-based image restoration," *IEEE Trans. on Image Process*, vol. 16, no. 12, pp. 2980–2991, 2007.

[22] E. Cands, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted l1 minimization," Caltech University, Tech. Rep., 2007.

[23] K. Kreutz-Delgado, J. Murray, B. Rao, K. Engan, T. Lee, and T. Sejnowski, "Dictionary learning algorithms for sparse representation," *Neural Comp.*, vol. 15, pp. 349–396, 2003.

[24] J. Leeuw, "Block-relaxation algorithms in statistics," in Information Systems and Data Analysis, ed. H.H. Bock, W. Lenski and M. M. Richter, Berlin: Springer-Verlag, pp. 308-325, 1994.

[25] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Annual of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[26] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Comm. Pure Appl. Math*, vol. 57, pp. 1413–1541, 2004.

[27] K. Lange, D. Hunter, and I. Yang, "Optimization transfer using surrogate objective functions," *Journal of Computational and Graphical Statistics*, vol. 9, no. 1, pp. 1–20, 2000.

[28] Z. Zhang, J. Kwok, and D. Yeung, "Surrogate maximization/minimization algorithms and extensions," *Machine Learning*, vol. 69, no. 1, pp. 1–33, 2007.

[29] K. Lange, *Optimization*. Springer-Verlag, 2004.

[30] A. Lyapunov, *Stability of motion*. Academic Press, 1966.

[31] R. Horn and C. Johnson, *Matrix Analysis*. Cambridge University Press, 1985.

[32] M. Fornasier and H. Rauhut, "Iterative thresholding algorithms," to appear in Applied and Computational Harmonic Analysis, 2007.

[33] M. Elad, B. Matalon, and M. Zibulevsky, "Coordinate and subspace optimization methods for linear least squares with non-quadratic regularization," *Applied and Computational Harmonic Analysis*, vol. 23, no. 3, pp. 346–367, 2007.

[34] D. Donoho and J. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994.

[35] L. Landweber, "An iterative formula for fredholm integral equations of the first kind," *American Journal of Mathematics*, vol. 73, pp. 615–624, 1951.

[36] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK User's Guide http://www.netlib.org/lapack/lug/lapack_lug.html, Third Edition*. Society for Industrial and Applied Mathematics (SIAM), 1999.

[37] O. Divorra Escoda, L. Granai, and P. Vandergheynst, "On the use of a priori information for sparse signal approximations," *IEEE Trans. on Signal Processing*, vol. 54, no. 9, pp. 3468–3482, 2006.

[38] R. Rockafellar, *Convex Analysis*. Princeton University Press, 1970.

[39] M. Fornasier and H. Rauhut, "Recovery algorithms for vector valued data with joint sparsity constraints," to appear in SIAM Journal of Numerical Analysis, 2007.

[40] L. Daudet and B. Torresani, "Hybrid representations for audiophonic signal encoding," *Signal Processing, special issue on Coding Beyond Standards*, vol. 82, no. 11, pp. 1595–1617, 2002.

[41] "Sparsify 0.2," the University of Edinburgh, 2007.

[42] J. Dattorro, *Convex Optimization and Euclidean Distance Geometry*. Meboo Publishing, 2005 (v2007.09.17).

[43] W. Zangwill, *Nonlinear Programming: A Unified Approach*. Printice-Hall, 1969.

[44] J. Fiorot and P. Huard, "Composition and union of general algorithms of optimization," *Mathematical Programming Study*, vol. 10, pp. 69–85, 1979.

[45] B. Palka, *An Introduction to Complex Function Theory*. Springer, 1991.

Mehrdad Yaghoobi (S'98-M'09) received the BSc. and MSc. in electrical and biomedical engineering in 1999 and 2002 from the University of Tehran and Sharif University of Technology, respectively. He spent one year as a research assistant in the AICTC, Sharif University of Technology before starting his PhD at Queen Mary University of London, in December 2005. He moved to the University of Edinburgh to accompany his supervisor in April 2006. He is now persuing the PhD degree in the Institute for Digital Communications (IDCom) at the University of Edinburgh. His current research interests include sparse approximation, dictionary selection, compressed sensing and audio modelling/coding.

He is recipient of EPSRC studentship and the University of Edinburgh international tuition fee waiver.

Thomas Blumensath (S'02-M'06) received the BSc hons. degree in music technology from Derby University, Derby, U.K., in 2002 and his Ph.D. degree in electronic engineering from Queen Mary, University of London, U.K., in 2006. He is currently a postdoctoral research fellow in the Institute for Digital Communication at the University of Edinburgh. His research interests include mathematical and statistical methods in signal processing with a focus on sparse signal models and their application.

Mike E. Davies (M00) received the B.A. (Hons.) degree in engineering from Cambridge University, Cambridge, U.K., in 1989 and the Ph.D. degree in nonlinear dynamics and signal processing from University College London, London (UCL), U.K., in 1993. Mike Davies was awarded a Royal Society Research Fellowship in 1993 and was an Associate Editor for IEEE Transactions in Speech, Language and Audio Processing, 2003-2007.

He currently holds the Jeffrey Collins SHEFC funded chair in Signal and Image Processing at the University of Edinburgh. His current research interests include: sparse approximation, compressed sensing and their applications.