

Optimized Projections for Compressed Sensing via Direct Mutual Coherence Minimization

Canyi Lu^a, Huan Li^b, Zhouchen Lin^{b,c,1}

^a*Department of Electrical and Computer Engineering, National University of Singapore, Singapore*

^b*Key Laboratory of Machine Perception (MOE), School of EECS, Peking University, China*

^c*Cooperative Medianet Innovation Center, Shanghai Jiao Tong University, China*

Abstract

Compressed Sensing (CS) is a new data acquisition theory based on the existence of a sparse representation of a signal and a projected dictionary \mathbf{PD} , where $\mathbf{P} \in \mathbb{R}^{m \times d}$ is the projection matrix and $\mathbf{D} \in \mathbb{R}^{d \times n}$ is the dictionary. To recover the signal from a small number m of measurements, it is expected that the projected dictionary \mathbf{PD} is of low mutual coherence. Several previous methods attempt to find the projection \mathbf{P} such that the mutual coherence of \mathbf{PD} is low. However, they do not minimize the mutual coherence directly and thus they may be far from optimal. Their used solvers lack convergence guarantee and thus the quality of their solutions is not guaranteed. This work aims to address these issues. We propose to find an optimal projection matrix by minimizing the mutual coherence of \mathbf{PD} directly. This leads to a nonconvex nonsmooth minimization problem. We approximate it by smoothing, solve it by alternating minimization and prove the convergence of our algorithm. To the best of our knowledge, this is the first work which directly minimizes the mutual coherence of the projected dictionary and has convergence guarantee. Numerical experiments demonstrate that our method can recover sparse signals better than existing ones.

Keywords: mutual coherence minimization, compressed sensing, convergence guarantee

¹corresponding author

1. Introduction

Compressed Sensing (CS) [1, 2] is a new sampling/data acquisition theory asserting that one can exploit sparsity or compressibility when acquiring signals of interest. It shows that signals which have a sparse representation with respect to appropriate bases
5 can be recovered from a small number of measurements. A fundamental problem in CS is how to construct a measurement matrix such that the number of measurements is near minimal.

Consider a signal $\mathbf{x} \in \mathbb{R}^d$ which is assumed to have a sparse representation with respect to a fixed overcomplete dictionary $\mathbf{D} \in \mathbb{R}^{d \times n}$ ($d < n$). This can be described as

$$\mathbf{x} = \mathbf{D}\boldsymbol{\alpha}, \quad (1)$$

where $\boldsymbol{\alpha} \in \mathbb{R}^n$ is a sparse representation coefficient, i.e., $\|\boldsymbol{\alpha}\|_0 \ll n$. Here $\|\boldsymbol{\alpha}\|_0$ denotes the ℓ_0 -norm which counts the number of nonzero elements in $\boldsymbol{\alpha}$. The solution to problem (1) is not unique since $d < n$. To find an appropriate solution in the solution set of (1), we need to use some additional structures of \mathbf{D} and $\boldsymbol{\alpha}$. Considering that $\boldsymbol{\alpha}$ is sparse, we are interested in finding the sparsest representation coefficient $\boldsymbol{\alpha}$. This leads to the following sparse representation problem

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0, \text{ s. t. } \mathbf{x} = \mathbf{D}\boldsymbol{\alpha}. \quad (2)$$

However, the above problem is NP-hard [3] and thus is challenging to solve. Some algorithms, such as Basis Pursuit (BP) [4] and Orthogonal Matching Pursuit (OMP)
10 [5], can be used to find suboptimal solutions.

An interesting theoretical problem is that under what conditions the optimal solution to (2) can be computed. If the solution is computable, can it be exactly or approximately computed by BP or OMP? Some previous works answer the above questions based on the mutual coherence of the dictionary \mathbf{D} [6].

Definition 1. Given $\mathbf{D} = [\mathbf{d}_1, \dots, \mathbf{d}_n] \in \mathbb{R}^{d \times n}$, its mutual coherence is defined as the largest absolute and normalized inner product between different columns of \mathbf{D} , i.e.,

$$\mu(\mathbf{D}) = \max_{\substack{1 \leq i, j \leq n \\ i \neq j}} \frac{|\mathbf{d}_i^T \mathbf{d}_j|}{\|\mathbf{d}_i\| \|\mathbf{d}_j\|}.$$

15 The mutual coherence measures the highest correlation between any two columns of \mathbf{D} . It is expected to be as low as possible in order to find the sparsest solution to (2).

Theorem 1. [6, 7, 8] For problem (2), if $\boldsymbol{\alpha}$ satisfies

$$\|\boldsymbol{\alpha}\|_0 < \frac{1}{2} \left(1 + \frac{1}{\mu(\mathbf{D})} \right), \quad (3)$$

then the following results hold:

- $\boldsymbol{\alpha}$ is the solution to (2).
- $\boldsymbol{\alpha}$ is also the solution to the following convex ℓ_1 -minimization problem

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1, \text{ s. t. } \mathbf{x} = \mathbf{D}\boldsymbol{\alpha},$$

where $\|\boldsymbol{\alpha}\|_1 = \sum_i |\alpha_i|$ is the ℓ_1 -norm of $\boldsymbol{\alpha}$.

- 20 • $\boldsymbol{\alpha}$ can be obtained by OMP.

The above theorem shows that if the mutual coherence of \mathbf{D} is low enough, then the sparsest solution to (2) is computable. Thus, how to construct a dictionary \mathbf{D} with low mutual coherence is crucial in sparse coding. In CS, to reduce the number of measurements, we face a similar problem on the sensing matrix construction.

The theory of CS guarantees that a signal having a sparse representation can be recovered exactly from a small set of linear and nonadaptive measurements. This result suggests that it may be possible to sense sparse signals by taking far fewer measurements than what the conventional Nyquist-Shannon sampling theorem requires. But note that CS differs from classical sampling in several aspects. First, the sampling theory typically considers infinite-length and continuous-time signals. In contrast, CS is a mathematical theory that focuses on measuring finite-dimensional vectors in \mathbb{R}^n . Second, rather than sampling the signal at specific points in time, CS systems typically acquire measurements in the form of inner products between the signal and general test functions. At last, the ways to dealing with the signal recovery are different. Given the signal $\mathbf{x} \in \mathbb{R}^d$ in (1), CS suggests replacing these n direct samples with m indirect ones by measuring linear projections of \mathbf{x} defined by a proper projection or sensing matrix $\mathbf{P} \in \mathbb{R}^{m \times d}$, i.e.,

$$\mathbf{y} = \mathbf{P}\mathbf{x}, \quad (4)$$

such that $m \ll d$. It means that instead of sensing all n elements of the original signal \mathbf{x} , we can sense \mathbf{x} indirectly by its compressed form \mathbf{y} in a much smaller size m . Surprisingly, the original signal \mathbf{x} can be recovered from the observed \mathbf{y} by using the sparse representation in (1), i.e. $\mathbf{y} = \mathbf{PD}\boldsymbol{\alpha}$ with the sparsest $\boldsymbol{\alpha}$. Thus the reconstruction requires solving the following problem

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0, \text{ s. t. } \mathbf{y} = \mathbf{M}\boldsymbol{\alpha}, \quad (5)$$

where $\mathbf{M} = \mathbf{PD} \in \mathbb{R}^{m \times n}$ is called the effective dictionary. Problem (5) is also NP-hard. As suggested by Theorem 1, if the mutual coherence of \mathbf{PD} is low enough, then the solution $\boldsymbol{\alpha}$ to (5) is computable by OMP or by solving the following convex problem

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1, \text{ s. t. } \mathbf{y} = \mathbf{M}\boldsymbol{\alpha}. \quad (6)$$

25 Finally, the original signal \mathbf{x} can be reconstructed by $\mathbf{x} = \mathbf{D}\boldsymbol{\alpha}$. So it is expected to find a proper projection matrix \mathbf{P} such that $\mu(\mathbf{PD})$ is low. Furthermore, many previous works [9, 10] show that the required number of measurements for recovering the signal \mathbf{x} by CS can be reduced if $\mu(\mathbf{PD})$ is low.

In summary, the above discussions imply that by choosing an appropriate projection
 30 matrix \mathbf{P} such that $\mu(\mathbf{PD})$ is low enough, the true signal \mathbf{x} can be recovered with high probability by efficient algorithms. At the beginning, random projection matrices were shown to be good choices since their columns are incoherent with any fixed basis \mathbf{D} with high probability [11]. However, many previous works [9, 12, 10] show that well designed deterministic projection matrices can often lead to better performance
 35 of signal reconstruction than random projections do. In this work, we focus on the construction of deterministic projection matrices. We first give a brief review on some previous deterministic methods.

1.1. Related Work

In this work, we only consider the case that \mathbf{D} is fixed while \mathbf{P} can be changed.
 40 Our target is to find \mathbf{P} by minimizing $\mu(\mathbf{M})$, where $\mathbf{M} = \mathbf{PD}$. If each column of \mathbf{M} is normalized to have unit Euclidean length, then $\mu(\mathbf{M}) = \|\mathbf{G}\|_{\infty, \text{off}}$, where $\mathbf{G} = (g_{ij}) = \mathbf{M}^T \mathbf{M}$ is named as the Gram matrix and $\|\mathbf{G}\|_{\infty, \text{off}} = \max_{i \neq j} |g_{ij}|$ is the

largest off-diagonal element of $|\mathbf{G}|$. Several previous works used the Gram matrix to find the projection matrix \mathbf{P} [9, 12, 10]. We give a review on these methods in the following.

1.1.1. The Algorithm of Elad

The algorithm of Elad [9] considers minimizing the t -averaged mutual coherence defined as the average of the absolute and normalized inner products between different columns of \mathbf{M} which are above t , i.e.,

$$\mu_t(\mathbf{M}) = \frac{\sum_{1 \leq i, j \leq k, i \neq j} \chi_t(|g_{ij}|) |g_{ij}|}{\sum_{1 \leq i, j \leq k, i \neq j} \chi_t(|g_{ij}|)},$$

where $\chi_t(x)$ is the characteristic function defined as

$$\chi_t(x) = \begin{cases} 1, & \text{if } x \geq t, \\ 0, & \text{otherwise,} \end{cases}$$

and t is a fixed threshold which controls the top fraction of the matrix elements of $|\mathbf{G}|$ that are to be considered.

To find \mathbf{P} by minimizing $\mu_t(\mathbf{M})$, some properties of the Gram matrix $\mathbf{G} = \mathbf{M}^T \mathbf{M}$ are used. Assume that each column of \mathbf{M} is normalized to have unit Euclidean length. Then

$$\text{diag}(\mathbf{G}) = \mathbf{1}, \tag{7}$$

$$\text{rank}(\mathbf{G}) = m. \tag{8}$$

The work [9] proposed to minimize $\mu_t(\mathbf{M})$ by iteratively updating \mathbf{P} as follows. First, initialize \mathbf{P} as a random matrix and normalize each column of \mathbf{PD} to have unit Euclidean length. Second, shrink the elements of $\mathbf{G} = \mathbf{M}^T \mathbf{M}$ (where $\mathbf{M} = \mathbf{PD}$) by

$$g_{ij} = \begin{cases} \gamma g_{ij}, & \text{if } |g_{ij}| \geq t, \\ \gamma t \text{sign}(g_{ij}), & \text{if } t > |g_{ij}| \geq \gamma t, \\ g_{ij}, & \text{if } \gamma t > |g_{ij}|, \end{cases}$$

where $0 < \gamma < 1$ is a down-scaling factor. Third, apply SVD and reduce the rank of \mathbf{G} to be equal to m . At last, build the square root \mathbf{S} of \mathbf{G} : $\mathbf{S}^T \mathbf{S} = \mathbf{G}$, where $\mathbf{S} \in \mathbb{R}^{m \times n}$, and find $\mathbf{P} = \mathbf{S} \mathbf{D}^\dagger$, where \dagger denotes the Moore-Penrose pseudoinverse.

There are several limitations of the algorithm of Elad. First, it is suboptimal since the t -averaged mutual coherence $\mu_t(\mathbf{M})$ is different from the mutual coherence $\mu(\mathbf{M})$ which is our real target. Second, the proposed algorithm to minimize $\mu_t(\mathbf{M})$ has no convergence guarantee. So the quality of the obtained solution is not guaranteed. Third, the choices of two parameters, t and γ , are crucial for the signal recovery performance in CS. However, there is no guideline for their settings and thus in practice it is usually difficult to find their best choices.

1.1.2. The Algorithm of Duarte-Carajalino and Sapiro

The algorithm of Duarte-Carajalino and Sapiro [12] is not a method that is based on mutual coherence. It instead aims to find the sensing matrix \mathbf{P} such that the corresponding Gram matrix is as close to the identity matrix as possible, i.e.,

$$\mathbf{G} = \mathbf{M}^T \mathbf{M} = \mathbf{D}^T \mathbf{P}^T \mathbf{P} \mathbf{D} \approx \mathbf{I}, \quad (9)$$

where \mathbf{I} denotes the identity matrix. Multiplying both sides of the previous expression by \mathbf{D} on the left and \mathbf{D}^T on the right, it becomes

$$\mathbf{D} \mathbf{D}^T \mathbf{P}^T \mathbf{P} \mathbf{D} \mathbf{D}^T \approx \mathbf{D} \mathbf{D}^T. \quad (10)$$

Let $\mathbf{D} \mathbf{D}^T = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T$ be the eigen-decomposition of $\mathbf{D} \mathbf{D}^T$. Then (10) is equivalent to

$$\mathbf{\Lambda} \mathbf{V}^T \mathbf{P}^T \mathbf{P} \mathbf{V} \mathbf{\Lambda} = \mathbf{\Lambda}. \quad (11)$$

Define $\mathbf{\Gamma} = \mathbf{P} \mathbf{V}$. Then they finally formulate the following model w.r.t. $\mathbf{\Gamma}$

$$\min_{\mathbf{\Gamma}} \|\mathbf{\Lambda} - \mathbf{\Lambda} \mathbf{\Gamma}^T \mathbf{\Gamma} \mathbf{\Lambda}\|_F. \quad (12)$$

After solving the above problem, the projection matrix can be obtained as $\mathbf{P} = \mathbf{\Gamma} \mathbf{V}^T$.

However, usually the signal recovery performance of the algorithm of Duarte-Carajalino and Sapiro is not very good. The reason is that \mathbf{M} is overcomplete and the Gram matrix \mathbf{G} cannot be an identity matrix. In this case, simply minimizing the difference between the Gram matrix \mathbf{G} and the identity matrix does not imply a solution \mathbf{M} with low mutual coherence.

1.1.3. The Algorithm of Xu et al.

The algorithm of Xu et al. [10] is motivated by the well-known Welch bound [13]. For any $\mathbf{M} \in \mathbb{R}^{m \times n}$, the mutual coherence $\mu(\mathbf{M})$ is lower bounded, e.g.,

$$\mu(\mathbf{M}) \geq \sqrt{\frac{n-m}{m(n-1)}}. \quad (13)$$

The algorithm of Xu et al. aims to find \mathbf{M} such that the off-diagonal elements of $\mathbf{G} = \mathbf{M}^T \mathbf{M}$ approximate the Welch bound well. They proposed to solve the following problem

$$\begin{aligned} \min_{\mathbf{G}} \quad & \|\mathbf{G} - \mathbf{G}_\Lambda\|_F \\ \text{s.t.} \quad & \mathbf{G}_\Lambda = \mathbf{G}_\Lambda^T, \text{diag}(\mathbf{G}_\Lambda) = \mathbf{1}, \|\mathbf{G}_\Lambda\|_{\infty, \text{off}} \leq \mu_W, \end{aligned} \quad (14)$$

where $\mu_W = \sqrt{\frac{n-m}{m(n-1)}}$. The proposed iterative solver for the above problem is similar to the algorithm of Elad. The main difference is the shrinkage function used to control the elements of \mathbf{G} . See [10] for more details.

70 However, their proposed solver in [10] for (14) also lacks convergence guarantee. Another issue is that, for $\mathbf{M} \in \mathbb{R}^{m \times n}$, the Welch bound (13) is not tight when n is large. Actually, the equality of (13) can hold only when $n \leq \frac{m(m+1)}{2}$. This implies that the algorithm of Xu et al. is not optimal when $n > \frac{m(m+1)}{2}$.

Beyond the above three methods, there are also some other mutual coherence optimization based methods for the dictionary learning. For example, the work [14] proposes a joint sparse coding and incoherent dictionary learning model which shares a similar idea as the algorithm of Duarte-Carajalino and Sapiro [12]. The work [15] considers a model with hard constraint on the mutual coherence and sparsity and proposes a heuristic iterative projection solver. Greedy algorithms are proposed in [16, 17] to
80 find a sensing matrix for a dictionary that gives low cumulative coherence.

1.2. Contributions

There are at least two main issues in the previous methods reviewed above. First, none of them aims to find \mathbf{P} by directly minimizing $\mu(\mathbf{P}\mathbf{D})$ which is our real target. Thus the objectives of these methods are not optimal. For their obtained solutions \mathbf{P} ,
85 $\mu(\mathbf{P}\mathbf{D})$ is usually much larger than the Welch bound in (13). Second, the algorithms

of Elad and Xu et al. have no convergence guarantee and thus they may produce very different solutions given slightly different initializations. The convergence issue may limit their applications in CS.

To address the above issues, we develop Direct Mutual Coherence Minimization (DMCM) models. First, we show how to construct a low mutual coherence matrix \mathbf{M} by minimizing $\mu(\mathbf{M})$ directly. This leads to a nonconvex and nonsmooth problem. To solve our new problem efficiently, we first smooth the objective function such that its gradient is Lipschitz continuous. Then we solve the approximate problem by proximal gradient which has convergence guarantee. Second, inspired by DMCM, we propose a DMCM based Projection (DMCM-P) model which aims to find a projection \mathbf{P} by minimizing $\mu(\mathbf{PD})$ directly. To solve the nonconvex DMCM-P problem, we then propose an alternating minimization method and prove its convergence. Experimental results show that our DMCM-P achieves the lowest mutual coherence of \mathbf{PD} and also leads to the best signal recovery performance.

2. Low Mutual Coherence Matrix Construction

In this section, we show how to construct a matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$ with low mutual coherence $\mu(\mathbf{M})$ by DMCM. Assume that each column of \mathbf{M} is normalized to unit Euclidean length. Then we aim to find \mathbf{M} by the following DMCM model

$$\begin{aligned} \min_{\mathbf{M} \in \mathbb{R}^{m \times n}} \mu(\mathbf{M}) &= \|\mathbf{M}^T \mathbf{M}\|_{\infty, \text{off}} \\ \text{s. t. } \|\mathbf{M}_i\|_2 &= 1, \quad i = 1, \dots, n, \end{aligned} \quad (15)$$

where \mathbf{M}_i (or $(\mathbf{M})_i$) denotes the i -th column of \mathbf{M} . The above problem is equivalent to

$$\begin{aligned} \min_{\mathbf{M} \in \mathbb{R}^{m \times n}} f(\mathbf{M}) &= \|\mathbf{M}^T \mathbf{M} - \mathbf{I}\|_{\infty} \\ \text{s. t. } \|\mathbf{M}_i\|_2 &= 1, \quad i = 1, \dots, n, \end{aligned} \quad (16)$$

where $\|\mathbf{A}\|_{\infty} = \max_{i,j} |a_{ij}|$ denotes the ℓ_{∞} -norm of \mathbf{A} . Solving the above problem is not easy since it is nonconvex and its objective is nonsmooth. In general, due to the nonconvexity, the globally optimal solution to (16) is not computable. We instead consider finding a locally optimal solution with convergence guarantee.

First, to ease the problem, we adopt the smoothing technique in [18] to smooth the nonsmooth ℓ_∞ -norm in the objective of (16). By the fact that the ℓ_1 -norm is the dual norm of the ℓ_∞ -norm, the objective function in (16) can be rewritten as

$$f(\mathbf{M}) = \|\mathbf{M}^T \mathbf{M} - \mathbf{I}\|_\infty = \max_{\|\mathbf{V}\|_1 \leq 1} \langle \mathbf{M}^T \mathbf{M} - \mathbf{I}, \mathbf{V} \rangle,$$

where $\|\mathbf{V}\|_1 = \sum_{i,j} |v_{ij}|$ denotes the ℓ_1 -norm of \mathbf{V} . Since $\{\mathbf{V} \mid \|\mathbf{V}\|_1 \leq 1\}$ is a bounded convex set, we can define a proximal function $d(\mathbf{V})$ for this set, where $d(\mathbf{V})$ is continuous and strongly convex on this set. A natural choice of $d(\mathbf{V})$ is $d(\mathbf{V}) = \frac{1}{2} \|\mathbf{V}\|_F^2$, where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. Hence, we have the following smooth approximation of f defined in (16):

$$f_\rho(\mathbf{M}) = \max_{\|\mathbf{V}\|_1 \leq 1} \langle \mathbf{M}^T \mathbf{M} - \mathbf{I}, \mathbf{V} \rangle - \frac{\rho}{2} \|\mathbf{V}\|_F^2, \quad (17)$$

where $\rho > 0$ is a smoothing parameter. Note that the smooth function f_ρ can approximate the nonsmooth f with an arbitrary precision and it is easier to be minimized. Indeed, f and f_ρ have the following relationship

$$f_\rho(\mathbf{M}) \leq f(\mathbf{M}) \leq f_\rho(\mathbf{M}) + \rho\gamma,$$

where $\gamma = \max_{\mathbf{V}} \{\frac{1}{2} \|\mathbf{V}\|_F^2 \mid \|\mathbf{V}\|_\infty \leq 1\}$. For any $\epsilon > 0$, if we choose $\rho = \frac{\epsilon}{\gamma}$, then $|f(\mathbf{M}) - f_\rho(\mathbf{M})| \leq \epsilon$. This implies that if ρ is sufficiently small, then the difference between f and f_ρ can be very small. This motivates us to use f_ρ to replace f in (16) and thus we have the following relaxed problem

$$\begin{aligned} & \min_{\mathbf{M} \in \mathbb{R}^{m \times n}} f_\rho(\mathbf{M}) \\ & \text{s. t. } \|\mathbf{M}_i\|_2 = 1, \quad i = 1, \dots, n. \end{aligned} \quad (18)$$

As f_ρ can approximate f at an arbitrary precision, solving (18) can still be regarded as directly minimizing the mutual coherence. Problem (18) is easier to solve since $\nabla f_\rho(\mathbf{M}) = \mathbf{M}(\mathbf{V}^* + \mathbf{V}^{*T})$, where \mathbf{V}^* is the optimal solution to (17), is Lipschitz continuous. That is, for any $\mathbf{M}_1, \mathbf{M}_2 \in \mathbb{R}^{m \times n}$, there exists a constant $L = 1/\rho$ such that

$$\|\nabla f_\rho(\mathbf{M}_1) - \nabla f_\rho(\mathbf{M}_2)\|_F \leq L \|\mathbf{M}_1 - \mathbf{M}_2\|_F.$$

Algorithm 1 Solve (18) by Proximal Gradient algorithm.

Initialize: $k = 0, \mathbf{M}_k \in \mathbb{R}^{m \times n}, \rho > 0, \alpha = 0.99\rho, K > 0.$

Output: $\mathbf{M}^* = \text{PG}(\mathbf{M}_k, \rho).$

while $k < K$ **do**

1. Compute \mathbf{V}_k by solving (21);
2. Compute \mathbf{M}_{k+1} by solving (19);
3. $k = k + 1.$

end while

With the above property, problem (18) can be solved by the proximal gradient method which updates \mathbf{M} in the $(k + 1)$ -th iteration by

$$\begin{aligned} \mathbf{M}_{k+1} &= \arg \min_{\mathbf{M}} \langle \nabla f_{\rho}(\mathbf{M}_k), \mathbf{M} - \mathbf{M}_k \rangle + \frac{1}{2\alpha} \|\mathbf{M} - \mathbf{M}_k\|_F^2 \\ &= \arg \min_{\mathbf{M}} \frac{1}{2} \|\mathbf{M} - (\mathbf{M}_k - \alpha \nabla f_{\rho}(\mathbf{M}_k))\|_F^2 \\ &\text{s. t. } \|\mathbf{M}_i\|_2 = 1, i = 1, \dots, n, \end{aligned} \quad (19)$$

where $\alpha > 0$ is the step size. To guarantee convergence, it is required that $\alpha < \rho$. In this work, we simply set $\alpha = 0.99\rho$. The above problem has a closed form solution by normalizing each column of $\mathbf{M}_k - \alpha \nabla f_{\rho}(\mathbf{M}_k)$, i.e.,

$$(\mathbf{M}_{k+1})_i = \frac{(\mathbf{M}_k - \alpha \nabla f_{\rho}(\mathbf{M}_k))_i}{\|(\mathbf{M}_k - \alpha \nabla f_{\rho}(\mathbf{M}_k))_i\|_2}. \quad (20)$$

To compute $\nabla f_{\rho}(\mathbf{M}_k) = \mathbf{M}_k(\mathbf{V}_k + \mathbf{V}_k^T)$, where \mathbf{V}_k is optimal to (17) when $\mathbf{M} = \mathbf{M}_k$, one has to solve (17) which is equivalent to the following problem

$$\begin{aligned} \mathbf{V}_k &= \arg \min_{\mathbf{V}} \frac{1}{2} \|\mathbf{V} - (\mathbf{M}_k^T \mathbf{M}_k - \mathbf{I})/\rho\|_F, \\ &\text{s. t. } \|\mathbf{V}\|_1 \leq 1. \end{aligned} \quad (21)$$

105 Solving the above problem requires computing a proximal projection onto the ℓ_1 ball. This can be done efficiently by the method in [19].

Iteratively updating \mathbf{V} by (21) and \mathbf{M} by (19) leads to the Proximal Gradient (PG) algorithm for solving problem (18). We summarize the whole procedure of PG for (18)

in Algorithm 1. For the convergence guarantee, PG can be proved to be convergent.
110 But we omit its proof since we will introduce a more general solver and provide the
convergence guarantee in Section 3. For the per-iteration cost of Algorithm 1, there
are two main parts. For the update of \mathbf{M} by (19), we need to compute $\nabla_{\rho} f(\mathbf{M}_k) =$
 $\mathbf{M}_k(\mathbf{V}_k + \mathbf{M}_k^T)$ which costs $O(mn^2)$. For the update of \mathbf{V} by (21), we need to compute
 $\mathbf{M}_k^T \mathbf{M}_k$ which costs $O(mn^2)$. Thus, the per-iteration cost of Algorithm 1 is $O(m^2n +$
115 $mn^2)$.

Though PG is guaranteed to converge, the obtained suboptimal solution to (18)
may be far from optimal to problem (16) which is our original target. There are two
important factors which may affect the quality of the obtained solution by PG. First,
due to the nonconvexity of (18), the solution may be sensitive to the initialization of
120 \mathbf{M} . Second, the smoothing parameter $\rho > 0$ should be small so that the objective f_{ρ}
in (18) can well approximate the objective f in (16). However, if ρ is directly set to
a very small value, PG may decrease the objective function value of (18) very slowly.
This can be easily seen from the updating of \mathbf{M} in (19), where $\alpha < \rho$. To address the
above two issues, we use a continuation trick to find a better solution to (16) by solving
125 (18) with different initializations. Namely, we begin with a relatively large value of
 ρ and reduce it gradually. For each fixed ρ , we solve (18) by PG in Algorithm 1 and
use its solution as a new initialization of \mathbf{M} in PG. To achieve a better solution, we
repeat the above procedure T times or until ρ reaches a predefined small value ρ_{\min} .
We summarize the procedure of PG with the continuation trick in Algorithm 2.

130 Finally, we would like to emphasize some advantages of our DMCM model (16)
and the proposed solver. A main merit of our model (16) is that it minimizes the mutual
coherence $\mu(\mathbf{M})$ directly and thus the mutual coherence of its optimal solution can be
low. Though the optimal solution is in general not computable due to the nonconvexity
of (16), our proposed solver, which first smooths the objective and then minimizes
135 it by PG, has convergence guarantee. To the best of our knowledge, this is the first
work which directly minimizes the mutual coherence of a matrix with convergence
guarantee.

Algorithm 2 Solve (18) by PG with continuation trick.

Initialize: $\rho > 0, \alpha = 0.99\rho, \eta > 1, \mathbf{M}, t = 0, T > 0$.

while $t < T$ **do**

1. $\mathbf{M} = \text{PG}(\mathbf{M}, \rho)$ by calling Algorithm 1;
2. $\rho = \rho/\eta, \alpha = 0.99\rho$;
3. $t = t + 1$.

end while

3. Low Mutual Coherence Based Projection

In this section, we show how to find a projection matrix \mathbf{P} such that $\mu(\mathbf{PD})$ can be as low as possible. This is crucial for signal recovery by CS associated to problem (5). Similar to the DMCM model shown in (16), an ideal way is to minimize $\mu(\mathbf{PD})$ directly, i.e.,

$$\begin{aligned} \min_{\mathbf{P} \in \mathbb{R}^{m \times d}} & \|(\mathbf{PD})^T(\mathbf{PD}) - \mathbf{I}\|_{\infty} \\ \text{s. t. } & \|\mathbf{PD}_i\|_2 = 1, i = 1, \dots, n. \end{aligned} \quad (22)$$

However, the constraint of (22) is more complex than the one in (16), and thus (22) is much more challenging to solve. We instead consider an approximate model of (22) based on the following observation.

Theorem 2. *For any $\mathbf{M}_1, \mathbf{M}_2 \in \mathbb{R}^{m \times n}$, if $\mathbf{M}_1 \rightarrow \mathbf{M}_2$, then $\mu(\mathbf{M}_1) \rightarrow \mu(\mathbf{M}_2)$.*

It is easy to prove the above result by the definition of the mutual coherence of a matrix. The above theorem indicates that the difference of the mutual coherences of two matrices is small when the difference of two matrices is small. This motivates us to find \mathbf{M} such that $\mu(\mathbf{M})$ is low and the difference between \mathbf{M} and \mathbf{PD} is small. So we have the following approximate model of (22):

$$\begin{aligned} \min_{\mathbf{P} \in \mathbb{R}^{m \times d}, \mathbf{M} \in \mathbb{R}^{m \times n}} & \|\mathbf{M}^T \mathbf{M} - \mathbf{I}\|_{\infty} + \frac{1}{2\beta} \|\mathbf{M} - \mathbf{PD}\|_F^2 \\ \text{s. t. } & \|\mathbf{M}_i\|_2 = 1, i = 1, \dots, n, \end{aligned} \quad (23)$$

where $\beta > 0$ trades off $\mu(\mathbf{M})$ and the difference between \mathbf{M} and \mathbf{PD} . To distinguish from the DMCM model in (16), in this paper we name the above model as DMCM based Projection (DMCM-P).
145

Now we show how to solve (23). First, we smooth $\|\mathbf{M}^T\mathbf{M} - \mathbf{I}\|_\infty$ as $f_\rho(\mathbf{M})$ defined in (17). Then problem (23) can be approximated by the following problem with a smooth objective:

$$\begin{aligned} \min_{\mathbf{P}, \mathbf{M}} F(\mathbf{M}, \mathbf{P}) &= f_\rho(\mathbf{M}) + \frac{1}{2\beta} \|\mathbf{M} - \mathbf{PD}\|_F^2 \\ \text{s. t. } \|\mathbf{M}_i\|_2 &= 1, i = 1, \dots, n. \end{aligned} \quad (24)$$

When both ρ and β are small, f_ρ is very close to f . So is $\mu(\mathbf{PD})$ to $\mu(\mathbf{M})$ because $\|\mathbf{M} - \mathbf{PD}\|_F$ has to be small. Thus solving problem (24) can still be regarded as minimizing the mutual coherence directly. We propose to alternately update \mathbf{P} and \mathbf{M} to solve problem (24).

1. Fix $\mathbf{P} = \mathbf{P}_k$ and update \mathbf{M} by

$$\begin{aligned} &\mathbf{M}_{k+1} \\ &= \arg \min_{\mathbf{M}} \langle \nabla f_\rho(\mathbf{M}_k), \mathbf{M} - \mathbf{M}_k \rangle + \frac{1}{2\alpha} \|\mathbf{M} - \mathbf{M}_k\|_F^2 \\ &\quad + \frac{1}{2\beta} \|\mathbf{M} - \mathbf{P}_k\mathbf{D}\|_F^2 \\ &= \arg \min_{\mathbf{M}} \frac{1}{2} \left\| \mathbf{M} - \frac{\left(\frac{1}{\alpha} \mathbf{M}_k + \frac{1}{\beta} \mathbf{P}_k \mathbf{D} - \nabla f_\rho(\mathbf{M}_k) \right)}{\frac{1}{\alpha} + \frac{1}{\beta}} \right\|_F^2 \\ &\text{s. t. } \|\mathbf{M}_i\|_2 = 1, i = 1, \dots, n, \end{aligned} \quad (25)$$

150 where $\alpha > 0$ is a step size satisfying $\alpha < \rho$. Similar to (19), the above problem has a closed form solution. To compute $\nabla f_\rho(\mathbf{M}_k)$ in (25), we also need to compute \mathbf{V}_k by solving (21).

2. Fix $\mathbf{M} = \mathbf{M}_{k+1}$ and update \mathbf{P} by solving

$$\mathbf{P}_{k+1} = \underset{\mathbf{P}}{\operatorname{argmin}} \|\mathbf{M}_{k+1} - \mathbf{PD}\|_F^2, \quad (26)$$

which has a closed form solution $\mathbf{P} = \mathbf{M}_{k+1}\mathbf{D}^\dagger$.

155 Iteratively updating \mathbf{P} by (26) and \mathbf{M} by (25) leads to the Alternating Minimization (AM) method for (24). We summarize the whole procedure of AM in Algorithm 3. It

Algorithm 3 Solve (24) by Alternating Minimization.

Initialize: $k = 0, \mathbf{P}_k \in \mathbb{R}^{m \times d}, \mathbf{M}_k \in \mathbb{R}^{m \times n}, \rho > 0, \alpha = 0.99\rho, \beta > 0$.

Output: $\{\mathbf{P}^* \mathbf{M}^*\} = \text{AM}(\mathbf{M}_k, \mathbf{P}_k, \rho, \beta)$.

while $k < K$ **do**

1. Compute \mathbf{V}_k by solving (21);
2. Compute \mathbf{M}_{k+1} by solving (25);
3. Compute \mathbf{P}_{k+1} by solving (26);
4. $k = k + 1$.

end while

can be easily seen that the per-iteration cost of Algorithm 3 is $O((d+m)n^2 + n^3)$. We can prove that the sequence generated by AM converges to a critical point.

We define

$$h(\mathbf{M}) = \begin{cases} 0, & \text{if } \|\mathbf{M}_i\|_2 = 1, i = 1, \dots, n, \\ +\infty, & \text{otherwise.} \end{cases} \quad (27)$$

¹⁶⁰ **Theorem 3.** Assume that \mathbf{D} in problem (24) is of full row rank. Let $\{(\mathbf{M}_k, \mathbf{P}_k)\}$ be the sequence generated by Algorithm 3. Then the following results hold:

(i) There exists some constants $a > 0$ and $b > 0$ such that

$$\begin{aligned} & h(\mathbf{M}_{k+1}) + F(\mathbf{M}_{k+1}, \mathbf{P}_{k+1}) \\ & \leq h(\mathbf{M}_k) + F(\mathbf{M}_k, \mathbf{P}_k) - a\|\mathbf{M}_{k+1} - \mathbf{M}_k\|_F^2 - b\|\mathbf{P}_{k+1} - \mathbf{P}_k\|_F^2. \end{aligned} \quad (28)$$

(ii) There exists $\mathbf{W}_{k+1} \in \nabla_{\mathbf{M}} F(\mathbf{M}_{k+1}, \mathbf{P}_{k+1}) + \partial h(\mathbf{M}_{k+1})$ and constants $c > 0, d > 0$, such that

$$\|\mathbf{W}_{k+1}\|_F \leq c\|\mathbf{M}_{k+1} - \mathbf{M}_k\|_F + d\|\mathbf{P}_k - \mathbf{P}_{k+1}\|_F, \quad (29)$$

$$\nabla_{\mathbf{P}} F(\mathbf{M}_{k+1}, \mathbf{P}_{k+1}) = \mathbf{0}. \quad (30)$$

(iii) There exist a subsequence $\{(\mathbf{M}_{k_j}, \mathbf{P}_{k_j})\}$ and $(\mathbf{M}^*, \mathbf{P}^*)$ such that $(\mathbf{M}_{k_j}, \mathbf{P}_{k_j}) \rightarrow (\mathbf{M}^*, \mathbf{P}^*)$ and $F(\mathbf{M}_{k_j}, \mathbf{P}_{k_j}) + h(\mathbf{M}_{k_j}) \rightarrow F(\mathbf{M}^*, \mathbf{P}^*) + h(\mathbf{M}^*)$.

165 The proof of Theorem 3 can be found in Appendix. Note that to guarantee the convergence of Algorithm 3, Theorem 3 requires \mathbf{D} in problem (24) to be of full row rank. Such an assumption usually holds in CS since $\mathbf{D} \in \mathbb{R}^{d \times n}$ is an overcomplete dictionary with $d < n$.

Based on Theorem 3, we then have the following convergence results.

Theorem 4. (Convergence to a critical point). *The sequence $\{(\mathbf{M}_k, \mathbf{P}_k)\}$ generated by Algorithm 3 converges to a critical point of $F(\mathbf{M}, \mathbf{P}) + h(\mathbf{M})$. Moreover, the sequence $\{(\mathbf{M}_k, \mathbf{P}_k)\}$ has a finite length, i.e.,*

$$\sum_{k=0}^{+\infty} (a \|\mathbf{M}_{k+1} - \mathbf{M}_k\| + b \|\mathbf{P}_{k+1} - \mathbf{P}_k\|) < \infty,$$

170 where $a > 0$ and $b > 0$ are constants as in Theorem 3 (i).

Theorem 4 is directly obtained by Theorem 2.9 in [20] based on the results in Theorem 3. Though AM is guaranteed to converge, the obtained solution to (24) may be far from optimal to problem (23) which is our original target. In order for (24) to approximate (23) well, $\rho > 0$ should be small. On the other hand, $\beta > 0$ should also
175 to be small such that the difference between \mathbf{M} and \mathbf{PD} is small and thus $\mu(\mathbf{PD})$ can well approximate $\mu(\mathbf{M})$. Similar to Algorithm 2, we use a continuation trick to achieve a good solution to (23). Namely, we begin with a relatively large value of $\rho > 0$ and $\beta > 0$ and reduce them gradually. For each fixed pair (ρ, β) , we solve (24) by AM in Algorithm 3 and use its solution as a new initialization of \mathbf{P} and \mathbf{M} in AM. We repeat
180 the procedure T times or until ρ and β reach predefined small values ρ_{\min} and β_{\min} . We summarize the procedure of AM with the continuation trick in Algorithm 4.

Finally, we would like to emphasize some advantages of our DMCM-P over previous methods. The main merit of our DMCM-P is that it is the first model which minimizes $\mu(\mathbf{PD})$ directly and the proposed solver also has convergence guarantee.
185 The algorithms of Elad [9] and Xu et al. [10] are also mutual coherence based methods. But their objectives are suboptimal and their solvers lack convergence guarantee.

Algorithm 4 Solve (24) by AM with continuation trick.

Initialize: $\rho > 0, \alpha = 0.99\rho, \beta > 0, \eta > 1, \mathbf{M}, \mathbf{P}, t = 0, T > 0.$

while $t < T$ **do**

1. $(\mathbf{P}, \mathbf{M}) = \text{AM}(\mathbf{P}, \mathbf{M}, \rho, \beta)$ by calling Algorithm 3;
2. $\rho = \rho/\eta, \alpha = 0.99\rho;$
3. $\beta = \beta/\eta;$
4. $t = t + 1.$

end while

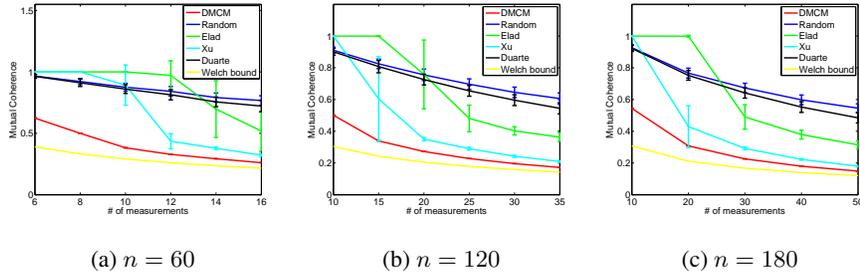


Figure 1: Plots of the means and standard deviations of mutual coherences of \mathbf{M} v.s. the number m of measurements.

It is worth mentioning that the sparse signal recovery can be guaranteed under some other different settings and conditions. The low mutual coherence property still plays an important role. For example, a similar recovery bound can be obtained under the additional assumption that the signs of the non-zero entries of the signal are chosen at random [21, 22]. The theory requires incoherence between the sensing and sparsity bases. The variable density sampling is a technique to recover the signal of highest sparsity by optimizing the sampling profile [23]. The proposed technique which directly minimizes the mutual coherence may be also applied in the variable density sampling to improve the recovery performance.

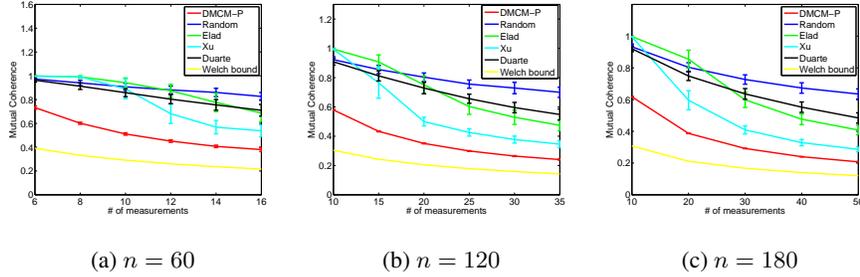


Figure 2: Plots of the means and standard deviations of mutual coherences of \mathbf{PD} v.s. the number m of measurements, where \mathbf{D} is a standard Gaussian random matrix.

4. Numerical Results

In this section, we conduct several experiments to verify the effectiveness of our proposed methods by comparing them with previous methods. The experiments consist of two parts. The first part shows the values of mutual coherence. The second part shows the signal recovery errors in CS.

4.1. Comparing the Mutual Coherence

This subsection presents two experiments to show the effectiveness of DMCM and DMCM-P, respectively. In the first experiment, we show that our DMCM is able to construct a matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$ with lower mutual coherence than previous methods do. We compare DMCM with

- Random: random matrix whose elements are drawn independently from the standard normal distribution.
- Elad: the algorithm of Elad [9] with $\mathbf{D} = \mathbf{I}$.
- Xu: the algorithm of Xu et al. [10] with $\mathbf{D} = \mathbf{I}$.
- Duarte: the algorithm of Duarte-Carjalino and Sapiro [12] with $\mathbf{D} = \mathbf{I}$.
- Welch bound: the Welch bound [13] shown in (13).

Note that the compared algorithms of Elad [9], Xu et al. [10] and Duarte-Carajalino and Sapiro [12] were designed to find a projection \mathbf{P} such that $\mathbf{M} = \mathbf{PD}$ has low mutual coherence. They can still be compared with our DMCM by setting \mathbf{D} as the identity matrix \mathbf{I} .

To solve our DMCM model in (18), we run Algorithm 2 for 15 iterations and Algorithm 1 for 1000 iterations. In Algorithm 2, we set $\rho_0 = 0.5$ and $\eta = 1.2$. \mathbf{M} is initialized as a Gaussian random matrix. In the method of Elad, we follow [9] to set $t = 0.2$ and $\gamma = 0.95$. In the method of Xu, we try multiple choices of the convex combination parameter α and set it as 0.5 which results in the lowest mutual coherence in most cases. The method of Duarte do not need special parameters. All the compared methods have the same random initializations of \mathbf{P} (except Duarte, which has a closed form solution).

The compared methods are tested on three settings with different sizes of $\mathbf{M} \in \mathbb{R}^{m \times n}$: (1) $m = [6 : 2 : 16], n = 60$; (2) $m = [10 : 5 : 35], n = 120$; and (3) $m = [10 : 10 : 50], n = 180$. Note that the constructed matrices may not be the same for the compared methods with different initializations. So for each choice of size (m, n) , we repeat the experiment for 100 times and record the means and standard deviations of the mutual coherences of the constructed matrices \mathbf{M} . The means and standard deviations of mutual coherences v.s. the number m of measurements are shown in Figure 1. It can be seen that the matrix constructed by our DMCM achieves much lower mutual coherences than previous methods do. The main reason is that our DMCM minimizes the mutual coherence of \mathbf{M} directly, while the objectives of all the previous methods are indirect. It can also be seen that the standard deviations of our method is close to zero, while some other compared methods may not be stable in some cases. A possible reason is that the solver of our method has convergence guarantee, while other methods do not.

For the second experiment in this subsection, we show that for given $\mathbf{D} \in \mathbb{R}^{d \times n}$ our DMCM-P is able to compute a projection $\mathbf{P} \in \mathbb{R}^{m \times d}$ such that $\mathbf{PD} \in \mathbb{R}^{m \times n}$ has low mutual coherence. We choose \mathbf{D} to be a Gaussian random matrix in this experiment. To solve our DMCM-P model in (23), we run Algorithm 4 for 15 iterations and Algorithm 3 for 1000 iterations. In Algorithm 4, we set $\rho_0 = 0.5, \beta = 2$ and $\eta = 1.2$. \mathbf{P} is

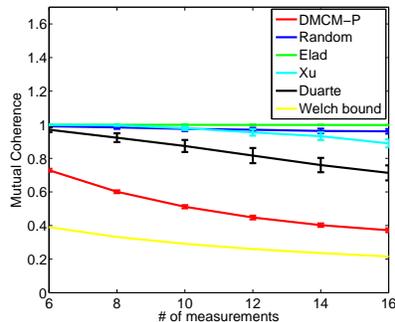


Figure 3: Plots of the means and standard deviations of mutual coherences of \mathbf{PD} v.s. the number m of measurements, where the elements of \mathbf{D} are uniformly distributed in $[0, 1]$.

Table 1: Comparison of running time (in seconds) of DMCM-P, Elad, Xu and Duarte on problem (23) under different settings.

	DMCM-P	Elad	Xu	Duarte
$m = 10, d = 30, n = 60$	181	5	5	0.0033
$m = 20, d = 60, n = 120$	582	8	8	0.004
$m = 30, d = 90, n = 180$	838	14	12	0.004

initialized as a Gaussian random matrix.

We compare our DMCM-P with the algorithms of Elad [9], Xu et al. [10] and
245 Duarte-Carajalino and Sapiro [12] on the mutual coherence of \mathbf{PD} . We test on three
settings: (1) $m = [6 : 2 : 16]$, $n = 60$, $d = 30$; (2) $m = [10 : 5 : 35]$, $n = 120$, $d = 60$;
and (3) $m = [10 : 10 : 50]$, $n = 180$, $d = 90$. Figure 2 shows the mutual coherence of
 \mathbf{PD} as a function of the number m of measurements. It can be seen that our DMCM-P
250 achieves the best projection such that \mathbf{PD} has the lowest mutual coherences in all the
three settings. So are the standard deviations. Note that our algorithm does not use any
special property of \mathbf{D} . So it is expected to work for \mathbf{D} in other distributions as well.
We test our method in the case that the elements of \mathbf{D} are uniformly distributed in $[0, 1]$

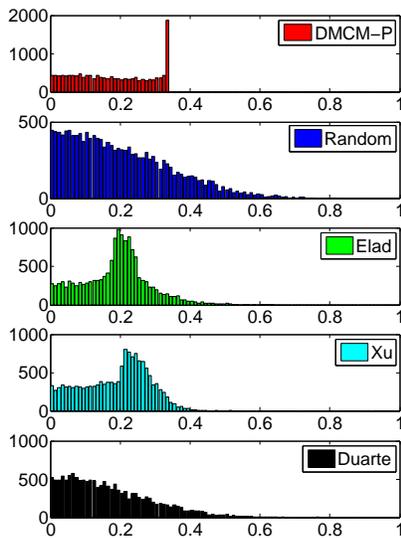


Figure 4: Distributions of the absolute values of $(\mathbf{PD})^T(\mathbf{PD})$.

and report the results in Figure 3. It can be seen that our method still outperforms other methods in both mean and standard deviation.

255 Furthermore, Figure 4 shows the distribution of the absolute values of inner products between distinct columns of \mathbf{PD} with $m = 20$, $n = 120$, and $d = 60$. It can be seen that our DMCM-P has the shortest tail, showing that the number of elements in the Gram matrix that are closer to the ideal Welch bound is larger than the compared methods. Such a result is consistent with the lowest mutual coherences shown in Figure
 260 2.

Finally, we report the running time of the algorithms of Elad, Xu, Duarte and our DMCM-P in Table 1. The settings of the algorithms are the same as those in Figure 2 and the running time is reported based on different choices of m , d and n . It can be seen that Duarte is the fastest method since it has a closed form solution. Our DMCM-P
 265 P is not very efficient since we use the continuation trick in Algorithm 4, which repeats

Algorithm 3 many times. Note that speeding up the algorithm, although valuable, is not the main focus of this paper. Actually, for many applications the projection matrix \mathbf{P} can be computed offline. So we leave the speeding-up issue as future work.

4.2. Comparing the CS Performance

270 In this subsection, we apply the optimized projection by our DMCM-P to CS. We first generate a T -sparse vector $\boldsymbol{\alpha} \in \mathbb{R}^n$, which constitutes a sparse representation of signal $\mathbf{x} = \mathbf{D}\boldsymbol{\alpha}$, where $\mathbf{x} \in \mathbb{R}^d$. The locations of nonzeros are chosen randomly and their values obey a uniform distribution in $[-1, 1]$. We choose the dictionary $\mathbf{D} \in \mathbb{R}^{d \times n}$ as a Gaussian random matrix, the DCT matrix and the matrix learned by K-SVD, 275 respectively. Then we apply different projection matrices \mathbf{P} learned by our DMCM-P, random projection matrix, and the algorithms of Elad [9], Xu et al. [10] and Duarte-Carajalino and Sapiro [12] to generate the compressed \mathbf{y} via $\mathbf{y} = \mathbf{P}\mathbf{D}\boldsymbol{\alpha}$. At last, we solve problem (5) by OMP to obtain $\hat{\boldsymbol{\alpha}}$. We compare the performance of projection matrices computed by different methods using the relative reconstruction error $\|\mathbf{x} - \mathbf{x}^*\|_2 / \|\mathbf{x}^*\|_2$ and the support recovery rate $|\text{support}(\mathbf{x}) \cap \text{support}(\mathbf{x}^*)| / |\text{support}(\mathbf{x}^*)|$, 280 where \mathbf{x}^* is the ground truth. A smaller reconstruction error and larger support recovery rate mean better CS performance.

We conduct two experiments in this subsection. The first one changes the number m of measurements and the second one changes the sparsity level T . For every value of 285 the aforementioned parameters we perform 3000 experiments and calculate the average relative reconstruction error and support recovery rate.

In the first experiment, we vary m and set $n = 60$, $d = 30$, $T = 2$ when \mathbf{D} is the Gaussian random matrix, $n = 60$, $d = 60$, $T = 2$ when \mathbf{D} is the DCT matrix and $n = 100$, $d = 100$, $T = 4$ when \mathbf{D} is the matrix learned by K-SVD, respectively. Figure 5, 290 6 and 7 show the average relative reconstruction error (left) and support recovery rate (right) v.s. the number m of measurements (T is fixed). In the last case, we follow [24] to train a dictionary for sparsely representing patches of size 10×10 extracted from the image Barbara. This image is of size 512×512 and thus has 253009 possible patches, considering all overlaps. We extract one tenth of these patches (uniformly spread) to 295 train on using the K-SVD with 50 iterations. The CS performance improves as m

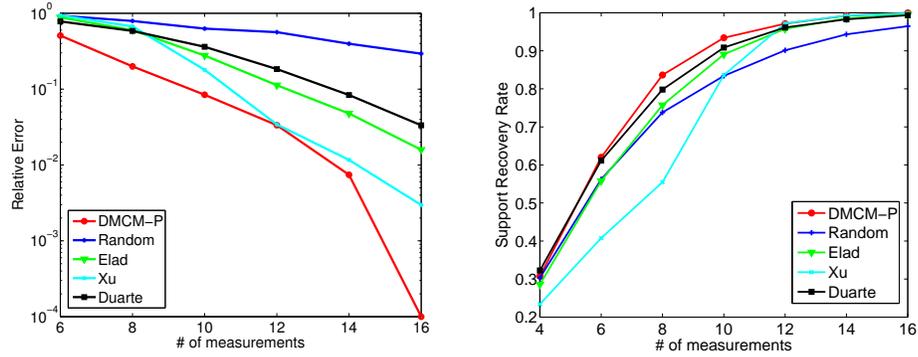


Figure 5: Signal reconstruction errors and support recovery rate v.s. number of measurements, where \mathbf{D} is the Gaussian random matrix.

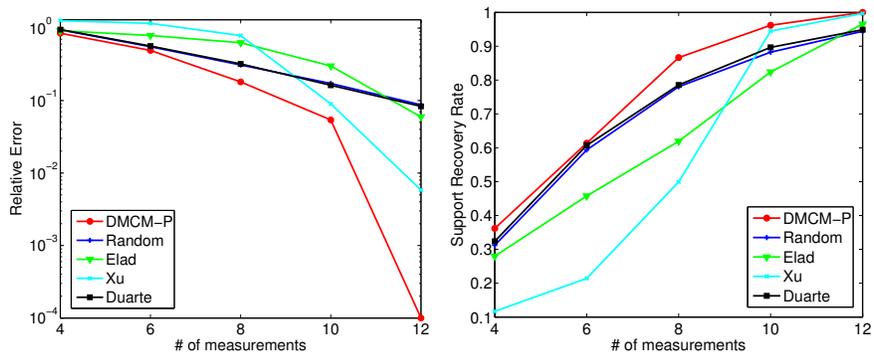


Figure 6: Signal reconstruction error and support recovery rate v.s. number of measurements, where \mathbf{D} is the DCT matrix.

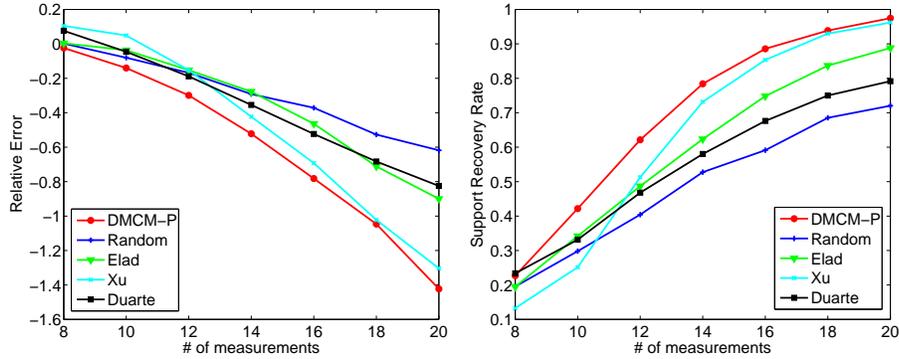


Figure 7: Signal reconstruction error and support recovery rate v.s. number of measurements, where \mathbf{D} is learned by K-SVD.

increases. Also, as expected, all the optimized projection matrices produce better CS performance than the random projection does, and our proposed DMCM-P consistently outperforms the algorithms of Elad, Xu et al. and Duarte-Carajalino and Sapiro.

In the second experiment, we vary the sparsity level T and set $m = 18$, $n = 180$ and $d = 90$ when \mathbf{D} is the Gaussian random matrix, $m = 15$, $n = 180$ and $d = 180$ when \mathbf{D} is the DCT matrix and $m = 12$, $n = 100$ and $d = 100$ when \mathbf{D} is the matrix learned by K-SVD. Figure 8, 9 and 10 show the average relative reconstruction error and support recovery rate as a function of the sparsity level T (m is fixed). The CS performance also improves as T decreases. Also, our DMCM-P consistently outperforms random projection and other deterministic projection optimization methods. This is due to the low mutual coherence of \mathbf{PD} thanks to our optimized projection method as verified in the previous experiments.

We also test the noisy case. We add Gaussian random noise with 0 mean and 0.01 variance to each element of the observation \mathbf{y} and then recover the true signal from this noisy \mathbf{y} . This time we test with \mathbf{D} in another different distribution and another choice of the ratio n/d . We generate elements of \mathbf{D} by a uniform distribution on $[0,1]$. We choose $m = [6 : 2 : 16]$, $d = 40$ and $n = 60$. Besides the sensing matrices constructed via optimization, we also compare DMCM-P with the the random binary matrix and Fourier matrix with random selected rows. Figure 11 shows the performance compari-

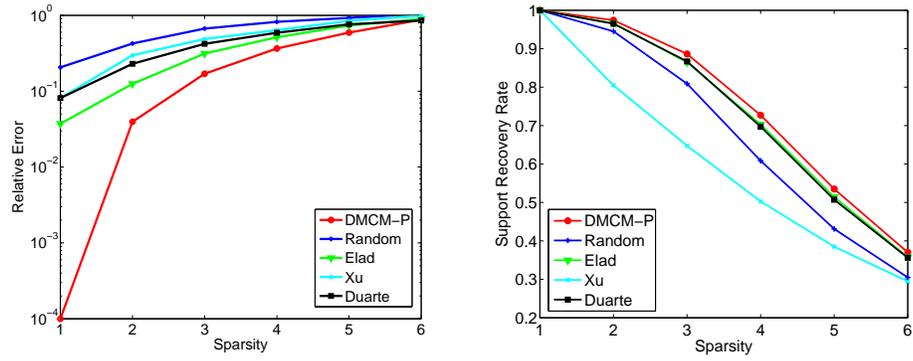


Figure 8: Signal reconstruction error and support recovery rate v.s. sparsity, where D is the Gaussian random matrix.

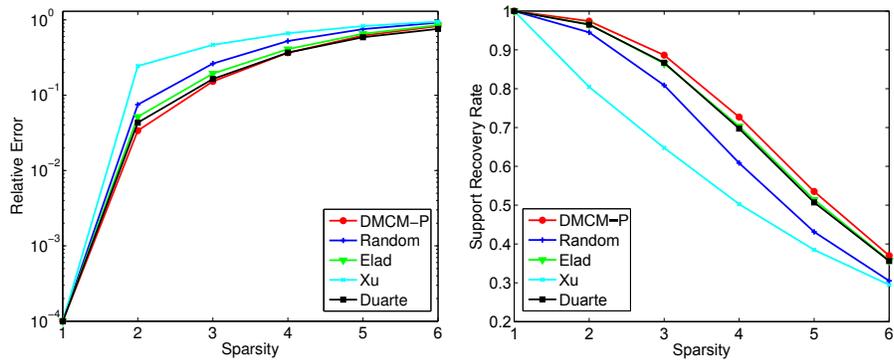


Figure 9: Signal reconstruction error and support recovery rate v.s. sparsity, where D is the DCT matrix.

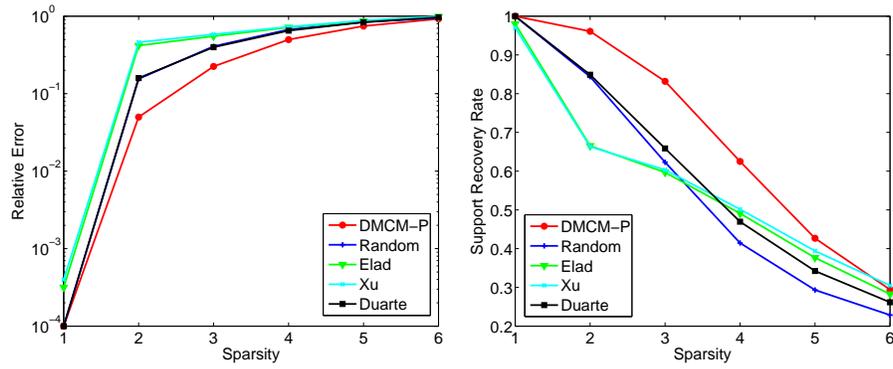


Figure 10: Signal reconstruction error and support recovery rate v.s. sparsity, where D is learned by K-SVD.

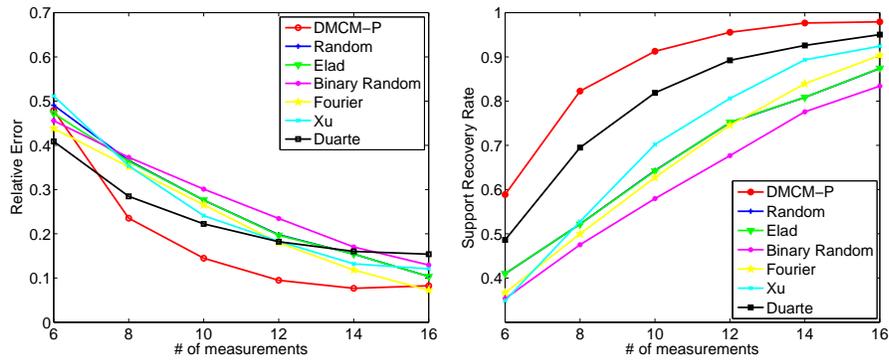


Figure 11: Signal reconstruction error and support recovery rate v.s. measurement in the noisy case, where D is the Gaussian random matrix.

315 son based on the relative reconstruction error and support recovery rate v.s. the number
of measurements. It can be seen that our method also achieves the best performance
in almost all cases. The improvement of our method over the random sensing matrices
(using Fourier matrix with random selected rows or the random binary matrices) are
significant.

320 5. Conclusions

This paper focuses on optimizing the projection matrix in CS for reconstructing
signals which are sparse in some overcomplete dictionary. We develop the first model
which aims to find a projection \mathbf{P} by minimizing the mutual coherence of \mathbf{PD}
directly. We solve the nonconvex problem by alternating minimization and prove the
325 convergence. Simulation results show that our method does achieve much lower mutu-
al coherence of \mathbf{PD} , and also leads to better CS performance. Considering that mutual
coherence is important in many applications besides CS, we expect that the proposed
construction will be useful in many other applications as well, besides CS.

There is some interesting future work. First, though we give the first solver with
330 convergence guarantee in Algorithm 1 for (16), the obtained solution is not guaranteed
to be globally optimal due to the nonconvexity of the problem. It is interesting to inves-
tigate when the obtained solution is globally optimal. Second, currently the proposed
method is not efficient, and it is valuable to find faster solvers. For example, we may
consider solving (16) and (22) by Alternating Direction Method of Multiplier (ADM-
335 M) after introducing some auxiliary variables, which may be more efficient than our
current solvers. But proving its convergence for nonconvex problems, (16) and (22),
will be challenging.

Acknowledgement

Z. Lin was supported by National Basic Research Program of China (973 Pro-
340 gram) (grant no. 2015CB352502), National Natural Science Foundation (NSF) of Chi-
na (grant nos. 61625301 and 61731018), Qualcomm, and Microsoft Research Asia.

Appendix

In this section, we give the proof of Theorem 3.

Definition 2. [25, 26] Let g be a proper and lower semicontinuous function.

1. For a given $\mathbf{x} \in \text{dom } g$, the Frechét subdifferential of g at \mathbf{x} , written as $\hat{\partial}g(\mathbf{x})$, is the set of all vectors $\mathbf{u} \in \mathbb{R}^n$ which satisfies

$$\liminf_{\mathbf{y} \neq \mathbf{x}, \mathbf{y} \rightarrow \mathbf{x}} \frac{g(\mathbf{y}) - g(\mathbf{x}) - \langle \mathbf{u}, \mathbf{y} - \mathbf{x} \rangle}{\|\mathbf{y} - \mathbf{x}\|} \geq 0.$$

2. The limiting-subdifferential, or simply the subdifferential, of g at $\mathbf{x} \in \mathbb{R}^n$, written as $\partial g(\mathbf{x})$, is defined through the following closure process

$$\begin{aligned} \partial g(\mathbf{x}) &:= \{ \mathbf{u} \in \mathbb{R}^n : \exists \mathbf{x}_k \rightarrow \mathbf{x}, g(\mathbf{x}_k) \rightarrow g(\mathbf{x}), \\ &\quad \mathbf{u}_k \in \hat{\partial}g(\mathbf{x}_k) \rightarrow \mathbf{u}, k \rightarrow \infty \}. \end{aligned}$$

Proposition 1. [25, 26] The following results hold:

1. In the nonsmooth context, the Fermat's rule remains unchanged: If $\mathbf{x} \in \mathbb{R}^n$ is a local minimizer of g , then $0 \in \partial g(\mathbf{x})$.
2. Let $(\mathbf{x}_k, \mathbf{u}_k)$ be a sequence such that $\mathbf{x}_k \rightarrow \mathbf{x}$, $\mathbf{u}_k \rightarrow \mathbf{u}$, $g(\mathbf{x}_k) \rightarrow g(\mathbf{x})$ and $\mathbf{u}_k \in \partial g(\mathbf{x}_k)$. Then $\mathbf{u} \in \partial g(\mathbf{x})$.
3. If f is a continuously differentiable function, then $\partial(f+g)(\mathbf{x}) = \nabla f(\mathbf{x}) + \partial g(\mathbf{x})$.

Proof of Theorem 3: First, (25) can be rewritten as

$$\begin{aligned} &\mathbf{M}_{k+1} \\ &= \arg \min_{\mathbf{M}} \langle \nabla f_\rho(\mathbf{M}_k), \mathbf{M} - \mathbf{M}_k \rangle + \frac{1}{2\alpha} \|\mathbf{M} - \mathbf{M}_k\|_F^2 \\ &\quad + \frac{1}{2\beta} \|\mathbf{M} - \mathbf{P}_k \mathbf{D}\|_F^2 + h(\mathbf{M}). \end{aligned}$$

By the optimality of \mathbf{M}_{k+1} , we have

$$\begin{aligned} &h(\mathbf{M}_{k+1}) + \langle \nabla f_\rho(\mathbf{M}_k), \mathbf{M}_{k+1} - \mathbf{M}_k \rangle \\ &\quad + \frac{1}{2\alpha} \|\mathbf{M}_{k+1} - \mathbf{M}_k\|_F^2 + \frac{1}{2\beta} \|\mathbf{M}_{k+1} - \mathbf{P}_k \mathbf{D}\|_F^2 \\ &\leq h(\mathbf{M}_k) + \frac{1}{2\beta} \|\mathbf{M}_k - \mathbf{P}_k \mathbf{D}\|_F^2. \end{aligned} \tag{31}$$

From the Lipschitz continuity of $\nabla f_\rho(\mathbf{M})$, we have

$$\begin{aligned}
& F(\mathbf{M}_{k+1}, \mathbf{P}_k) \\
&= f_\rho(\mathbf{M}_{k+1}) + \frac{1}{2\beta} \|\mathbf{M}_{k+1} - \mathbf{P}_k \mathbf{D}\|_F^2 \\
&\leq f_\rho(\mathbf{M}_k) + \langle \nabla f_\rho(\mathbf{M}_k), \mathbf{M}_{k+1} - \mathbf{M}_k \rangle \\
&\quad + \frac{1}{2\rho} \|\mathbf{M}_{k+1} - \mathbf{M}_k\|_F^2 + \frac{1}{2\beta} \|\mathbf{M}_{k+1} - \mathbf{P}_k \mathbf{D}\|_F^2.
\end{aligned} \tag{32}$$

Add (31) and (32), we have

$$\begin{aligned}
& h(\mathbf{M}_{k+1}) + F(\mathbf{M}_{k+1}, \mathbf{P}_k) \\
&\leq h(\mathbf{M}_k) + f_\rho(\mathbf{M}_k) - \left(\frac{1}{2\alpha} - \frac{1}{2\rho} \right) \|\mathbf{M}_{k+1} - \mathbf{M}_k\|_F^2 \\
&\quad + \frac{1}{2\beta} \|\mathbf{M}_k - \mathbf{P}_k \mathbf{D}\|_F^2 \\
&= h(\mathbf{M}_k) + F(\mathbf{M}_k, \mathbf{P}_k) - \left(\frac{1}{2\alpha} - \frac{1}{2\rho} \right) \|\mathbf{M}_{k+1} - \mathbf{M}_k\|_F^2.
\end{aligned} \tag{33}$$

Note that $F(\mathbf{M}_{k+1}, \mathbf{P}) = \frac{1}{2\beta} \|\mathbf{M}_{k+1} - \mathbf{P} \mathbf{D}\|_F^2$ is $\frac{1}{\beta} \sigma_{\min}^2(\mathbf{D})$ -strongly convex, where $\sigma_{\min}(\mathbf{D})$ denotes the smallest singular value of \mathbf{D} and it is positive since \mathbf{D} is of full rank. Then by Lemma B.5 in [27] and the optimality of \mathbf{P}_{k+1} to (26), we have

$$F(\mathbf{M}_{k+1}, \mathbf{P}_{k+1}) \leq F(\mathbf{M}_{k+1}, \mathbf{P}_k) - \frac{1}{2\beta} \sigma_{\min}^2(\mathbf{D}) \|\mathbf{P}_{k+1} - \mathbf{P}_k\|_F^2. \tag{34}$$

Combining (33) and (34) leads to

$$\begin{aligned}
& h(\mathbf{M}_{k+1}) + F(\mathbf{M}_{k+1}, \mathbf{P}_{k+1}) \\
&\leq h(\mathbf{M}_k) + F(\mathbf{M}_k, \mathbf{P}_k) - \left(\frac{1}{2\alpha} - \frac{1}{2\rho} \right) \|\mathbf{M}_{k+1} - \mathbf{M}_k\|_F^2 - \frac{1}{2\beta} \sigma_{\min}^2(\mathbf{D}) \|\mathbf{P}_{k+1} - \mathbf{P}_k\|_F^2.
\end{aligned} \tag{35}$$

Second, by the optimality of \mathbf{M}_{k+1} , we have

$$\begin{aligned}
0 \in & \partial h(\mathbf{M}_{k+1}) + \nabla f_\rho(\mathbf{M}_k) + \frac{1}{\alpha} (\mathbf{M}_{k+1} - \mathbf{M}_k) \\
& + \frac{1}{\beta} (\mathbf{M}_{k+1} - \mathbf{P}_k \mathbf{D}).
\end{aligned} \tag{36}$$

Thus, there exists $\mathbf{W}_{k+1} \in \nabla_{\mathbf{M}} F(\mathbf{M}_{k+1}, \mathbf{P}_{k+1}) + \partial h(\mathbf{M}_{k+1})$, such that

$$\begin{aligned} \mathbf{W}_{k+1} &\in \nabla f_{\rho}(\mathbf{M}_{k+1}) + \frac{1}{\beta}(\mathbf{M}_{k+1} - \mathbf{P}_{k+1}\mathbf{D}) + \partial h(\mathbf{M}_{k+1}) \\ &= \nabla f_{\rho}(\mathbf{M}_k) + \frac{1}{\beta}(\mathbf{M}_{k+1} - \mathbf{P}_k\mathbf{D}) + \partial h(\mathbf{M}_{k+1}) \\ &\quad + (f_{\rho}(\mathbf{M}_{k+1}) - f_{\rho}(\mathbf{M}_k)) + \frac{1}{\beta}(\mathbf{P}_k - \mathbf{P}_{k+1})\mathbf{D}. \end{aligned} \quad (37)$$

Then, combining (36) and (37) leads to

$$\begin{aligned} \|\mathbf{W}_{k+1}\|_F &\leq \left\| \nabla f_{\rho}(\mathbf{M}_k) + \frac{1}{\beta}(\mathbf{M}_{k+1} - \mathbf{P}_k\mathbf{D}) + \partial h(\mathbf{M}_{k+1}) \right\|_F \\ &\quad + \|f_{\rho}(\mathbf{M}_{k+1}) - f_{\rho}(\mathbf{M}_k)\|_F + \frac{1}{\beta} \|(\mathbf{P}_k - \mathbf{P}_{k+1})\mathbf{D}\|_F \\ &\leq \frac{1}{\alpha} \|\mathbf{M}_{k+1} - \mathbf{M}_k\|_F + \frac{1}{\rho} \|\mathbf{M}_{k+1} - \mathbf{M}_k\|_F + \frac{1}{\beta} \|\mathbf{D}\|^2 \|\mathbf{P}_k - \mathbf{P}_{k+1}\|_F, \end{aligned} \quad (38)$$

$$(39)$$

where (39) uses the property that $\nabla f_{\rho}(\mathbf{M})$ is Lipschitz continuous with the Lipschitz constant $1/\rho$. Also, by the optimality of \mathbf{P}_{k+1} , we have

$$\mathbf{0} = \nabla_{\mathbf{P}} F(\mathbf{M}_{k+1}, \mathbf{P}_{k+1}) = (\mathbf{M}_{k+1} - \mathbf{P}_{k+1}\mathbf{D})\mathbf{D}^T. \quad (40)$$

Third, note that $F(\mathbf{M}, \mathbf{P})$ is coercive, i.e., $F(\mathbf{M}, \mathbf{P})$ is bounded from below and $F(\mathbf{M}, \mathbf{P}) \rightarrow +\infty$ when $\|[\mathbf{M}, \mathbf{P}]\|_F \rightarrow +\infty$. It can be seen from (35) that $F(\mathbf{M}_k, \mathbf{P}_k)$ is bounded. Thus $\{\mathbf{M}_k, \mathbf{P}_k\}$ is bounded. Then there exists an accumulation point $(\mathbf{M}^*, \mathbf{P}^*)$ and a subsequence $\{\mathbf{M}_{k_j}, \mathbf{P}_{k_j}\}$ such that $(\mathbf{M}_{k_j}, \mathbf{P}_{k_j}) \rightarrow (\mathbf{M}^*, \mathbf{P}^*)$ as $j \rightarrow +\infty$. Since $F(\mathbf{M}, \mathbf{P})$ is continuously differentiable, we have $F(\mathbf{M}_{k_j}, \mathbf{P}_{k_j}) \rightarrow F(\mathbf{M}^*, \mathbf{P}^*)$. As $h(\mathbf{M}_k) = 0$ for all k and the set $\{\mathbf{M} : \|\mathbf{M}_i\|_2 = 1, i = 1, \dots, n\}$ is closed, we have $h(\mathbf{M}^*) = 0$ and $F(\mathbf{M}_{k_j}, \mathbf{P}_{k_j}) + h(\mathbf{M}_{k_j}) \rightarrow F(\mathbf{M}^*, \mathbf{P}^*) + h(\mathbf{M}^*)$. ■

Reference

- [1] E. J. Candès, J. Romberg, T. Tao, Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information, IEEE Transactions on Information Theory 52 (2) (2006) 489–509.
- [2] D. L. Donoho, Compressed sensing, IEEE Transactions on Information Theory 52 (4) (2006) 1289–1306.

- [3] B. K. Natarajan, Sparse approximate solutions to linear systems, *SIAM Journal on Computing* 24 (2) (1995) 227–234.
- 375 [4] S. S. Chen, D. L. Donoho, M. A. Saunders, Atomic decomposition by basis pursuit, *SIAM Journal on Scientific Computing* 20 (1) (1998) 33–61.
- [5] Y. C. Pati, R. Rezaifar, P. Krishnaprasad, Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition, in: *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, IEEE, 1993, pp. 40–44.
- 380 [6] R. Gribonval, M. Nielsen, Sparse representations in unions of bases, *IEEE Transactions on Information Theory* 49 (12) (2003) 3320–3325.
- [7] D. L. Donoho, M. Elad, Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization, *Proceedings of the National Academy of Sciences* 100 (5) (2003) 2197–2202.
- 385 [8] J. A. Tropp, Greed is good: Algorithmic results for sparse approximation, *IEEE Transactions on Information Theory* 50 (10) (2004) 2231–2242.
- [9] M. Elad, Optimized projections for compressed sensing, *IEEE Transactions on Signal Processing* 55 (12) (2007) 5695–5702.
- 390 [10] J. Xu, Y. Pi, Z. Cao, Optimized projection matrix for compressive sensing, *EURASIP Journal on Advances in Signal Processing* 2010 (2010) 43.
- [11] E. J. Candes, T. Tao, Near-optimal signal recovery from random projections: Universal encoding strategies?, *IEEE Transactions on Information Theory* 52 (12) (2006) 5406–5425.
- 395 [12] J. M. Duarte-Carvajalino, G. Sapiro, Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization, *IEEE Transactions on Image Processing* 18 (7) (2009) 1395–1408.
- [13] L. Welch, Lower bounds on the maximum cross correlation of signals (corresp.), *IEEE Transactions on Information theory* (1974) 397–399.

- 400 [14] C. Bao, Y. Quan, H. Ji, A convergent incoherent dictionary learning algorithm for sparse coding, in: European Conference on Computer Vision, Springer, 2014, pp. 302–316.
- [15] D. Barchiesi, M. D. Plumbley, Learning incoherent dictionaries for sparse approximation using iterative projections and rotations, *IEEE Transactions on Signal Processing* 61 (8) (2013) 2055–2065.
- 405 [16] B. Li, Y. Shen, J. Li, Dictionaries construction using alternating projection method in compressive sensing, *IEEE Signal Processing Letters* 18 (11) (2011) 663–666.
- [17] K. Schnass, P. Vandergheynst, Dictionary preconditioning for greedy algorithms, *IEEE Transactions on Signal Processing* 56 (5) (2008) 1994–2002.
- 410 [18] Y. Nesterov, Smooth minimization of non-smooth functions, *Mathematical Programming* 103 (1) (2005) 127–152.
- [19] Y. S. John Duchi, Shai Shalev-Shwartz, T. Chandra, Efficient projections onto the ℓ_1 -ball for learning in high dimensions, in: International Conference of Machine Learning, 2008.
- 415 [20] H. Attouch, J. Bolte, B. F. Svaiter, Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods, *Mathematical Programming* 137 (1-2) (2013) 91–129.
- 420 [21] H. Rauhut, Compressive sensing and structured random matrices, *Theoretical foundations and numerical methods for sparse recovery* 9 (2010) 1–92.
- [22] E. Candes, J. Romberg, Sparsity and incoherence in compressive sampling, *Inverse problems* 23 (3) (2007) 969.
- 425 [23] G. Puy, P. Vandergheynst, Y. Wiaux, On variable density compressive sampling, *IEEE Signal Processing Letters* 18 (10) (2011) 595–598.

- [24] M. Eald, *Sparse and redundant representations: From theory to applications in signal and image processing*, Springer.
- [25] R. R. Tyrrell, W. Roger (Eds.), *Variational Analysis*, Springer, 1998.
- [26] S. S. Jérôme Bolte, M. Teboullez, Proximal alternating linearized minimization
430 for nonconvex and nonsmooth problems, *Mathematical Programming* 146 (1-2)
(2014) 459–494.
- [27] J. Mairal, Optimization with first-order surrogate functions, in: *International Conference on Machine Learning*, 2013, pp. 783–791.