MATRIX ALPS: ACCELERATED LOW RANK AND SPARSE MATRIX RECONSTRUCTION

*Anastasios Kyrillidis and Volkan Cevher**

Laboratory for Information and Inference Systems, EPFL

ABSTRACT

We propose MATRIX ALPS for recovering a sparse plus low-rank decomposition of a matrix given its corrupted and incomplete linear measurements. Our approach is a first-order projected gradient method over non-convex sets, and it exploits a well-known memory-based acceleration technique. We theoretically characterize the convergence properties of MATRIX ALPS using the stable embedding properties of the linear measurement operator. We then numerically illustrate that our algorithm outperforms the existing convex as well as non-convex state-of-the-art algorithms in computational efficiency without sacrificing stability.

1. INTRODUCTION

Finding a low rank plus sparse matrix decomposition from a set of—possibly incomplete and noisy—measurements is critical in many applications. The list has expanded over the last ten years: examples include MRI signal processing, collaborative filtering, hyperspectral image analysis, large-scale data processing, etc. A general statement of the problem under consideration can be described as follows:

PROBLEM. *Given a linear operator $\boldsymbol{\mathcal{A}} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$ and a set of observations $\boldsymbol{y} \in \mathbb{R}^p$ (usually $p \ll m \times n$):*

$$\boldsymbol{y} = \boldsymbol{\mathcal{A}}\mathbf{X}^* + \boldsymbol{\varepsilon}, \tag{1}$$

where $\mathbf{X}^* := \mathbf{L}^* + \mathbf{M}^* \in \mathbb{R}^{m\times n}$ is the superposition of a rank-k $\mathbf{L}^*$ and a s-sparse $\mathbf{M}^*$ component that we desire to recover, identify a matrix $\widehat{\mathbf{L}} \in \mathbb{R}^{m\times n}$ of rank (at most) k and a matrix $\widehat{\mathbf{M}} \in \mathbb{R}^{m\times n}$ with sparsity level $\big\|\widehat{\mathbf{M}}\big\|_0 \leq s$ such that:

$$\big\{\widehat{\mathbf{L}},\ \widehat{\mathbf{M}}\big\} = \underset{\mathbf{L},\ \mathbf{M}\colon\ \mathrm{rank}(\mathbf{L})\leq k,\ \|\mathbf{M}\|_0\leq s}{\arg\min} \big\|\boldsymbol{y} - \boldsymbol{\mathcal{A}}(\mathbf{L}+\mathbf{M})\big\|_2. \tag{2}$$

Here, $\boldsymbol{\varepsilon} \in \mathbb{R}^p$ represents the potential noise term. For different linear operator $\boldsymbol{\mathcal{A}}$ and signal $\mathbf{X}^*$ configurations, the above problem arises in various research fields. Next, we briefly address some of the frameworks that (2) is involved.

1.1. Compressed sensing and affine rank minimization

In the standard Compressed Sensing (CS) framework, we desire to reconstruct a n-dimensional, s-sparse loading vector through a p-dimensional set of observations with $p \ll n$. This problem can be solved by finding the minimizer $\widehat{\mathbf{X}} := \widehat{\mathbf{M}}$ of:

$$\big\{\widehat{\mathbf{M}}\big\} = \underset{\mathbf{M}\ :\mathbf{M}\in\mathbb{D}^n,\ \|\mathbf{M}\|_0\leq s}{\arg\min} \big\|\boldsymbol{y} - \boldsymbol{\mathcal{A}}\mathbf{M}\big\|_2. \tag{3}$$

where we reserve $\mathbb{D}^n$ to denote the set of $n\times n$ diagonal matrices. To establish solution uniqueness and reconstruction stability in (3), $\boldsymbol{\mathcal{A}}$ is usually assumed to satisfy the *sparse restricted isometry property* (sparse-RIP) [1] where:

$$(1-\delta_s(\boldsymbol{\mathcal{A}}))\big\|\mathbf{X}\big\|_F \leq \big\|\boldsymbol{\mathcal{A}}\mathbf{X}\big\|_2 \leq (1+\delta_s(\boldsymbol{\mathcal{A}}))\big\|\mathbf{X}\big\|_F, \tag{4}$$

$\forall \mathbf{X} \in \mathbb{D}^n$ with $\big\|\mathbf{X}\big\|_0 \leq s$ and $\delta_s(\boldsymbol{\mathcal{A}}) \in (0,1)$.

In the general affine rank minimization (ARM) problem, we aim to recover a low-rank matrix $\mathbf{X}^* := \mathbf{L}^*$ from a set of observations $\boldsymbol{y} \in \mathbb{R}^p$, according to (1). The challenge is to reconstruct the true matrix given $p \ll m \cdot n$. A practical means to tackle this problem is by finding the simplest solution $\widehat{\mathbf{X}} := \widehat{\mathbf{L}}$ of minimum rank that minimizes the data error as:

$$\big\{\widehat{\mathbf{L}}\big\} = \underset{\mathbf{L}\colon\ \mathrm{rank}(\mathbf{L})\leq k}{\arg\min} \big\|\boldsymbol{y} - \boldsymbol{\mathcal{A}}\mathbf{L}\big\|_2. \tag{5}$$

[2] provides guarantees for exact and unique solution using the rank-RIP property for affine transformations where $\boldsymbol{\mathcal{A}}$ satisfies:

$$(1-\delta_k(\boldsymbol{\mathcal{A}}))\big\|\mathbf{X}\big\|_F \leq \big\|\boldsymbol{\mathcal{A}}\mathbf{X}\big\|_2 \leq (1+\delta_k(\boldsymbol{\mathcal{A}}))\big\|\mathbf{X}\big\|_F, \tag{6}$$

$\forall \mathbf{X} \in \mathbb{R}^{m\times n}$ with $\mathrm{rank}(\mathbf{X}) \leq k$ and $\delta_k(\boldsymbol{\mathcal{A}}) \in (0,1)$.

1.2. Fusing low-dimensional embedding models

Robust Principal Component Analysis (RPCA) deals with the challenge of recovering a low rank and a sparse matrix component from a *complete* data matrix. In mathematical terms, we acquire a finite set of observations $\mathbf{Y} \in \mathbb{R}^{m\times n}$ according to $\mathbf{Y} = \mathbf{L}^* + \mathbf{M}^*$ with $\mathbf{L}^* \in \mathbb{R}^{m\times n}$ and $\mathbf{M}^* \in \mathbb{R}^{m\times n}$, defined above. The "robust" characterization of the RPCA problem refers to $\mathbf{M}^*$ having *gross* non-zero entries with *arbitrary* energy. Under mild assumptions concerning the incoherence between $\mathbf{L}^*$ and $\mathbf{M}^*$ [3], we can efficiently reconstruct both the low-rank and sparse components using convex and non-convex optimization approaches [3, 4].

1.3. Contributions

While solving the RPCA problem itself is a difficult task, here we assume: (i) $\boldsymbol{\mathcal{A}}$ is an arbitrary linear operator satisfying both sparse- and rank-RIP (this assumption includes the identity linear map of RPCA as a special case) and, (ii) the total number of observations in $\boldsymbol{y}$ is much less compared to the total number of variables we want to recover, i.e., $p \ll m \cdot n$. Our contributions are two-fold:

- For noisy settings and arbitrary operator $\boldsymbol{\mathcal{A}}$ satisfying sparse- and rank-RIP, we provide better restricted isometry constant guarantees compared to state-of-the-art approaches [5].
- We introduce MATRIX ALPS, an accelerated, memory-based algorithm along with preliminary convergence analysis.

The organization of the paper is as follows. In Section 2, we describe the algorithms in a nutshell and present the main theorem of the paper in Section 3. In Section 4 we briefly study acceleration techniques in the recovery process. We provide empirical support for our claims for better data recovery performance and reduced complexity in Section 5.

*This work was supported in part by the European Commission under Grant MIRG-268398, ERC Future Proof, and DARPA KeCoM program #11-DARPA-1055. VC also would like to acknowledge Rice University for his Faculty Fellowship.

1: **Input:** $\boldsymbol{y}$, $\boldsymbol{\mathcal{A}}$, $\boldsymbol{\mathcal{A}}^*$, Tolerance η, MaxIterations
2: **Initialize:** $\{\mathbf{L}_0, \mathbf{M}_0\} \leftarrow 0$, $\{\mathcal{L}_0, \mathcal{M}_0\} \leftarrow \{\emptyset\}$, $i \leftarrow 0$
3: **repeat**
4: $\mathcal{S}_i^{\mathcal{L}} \leftarrow \mathcal{D}_i^{\mathcal{L}} \cup \mathcal{L}_i$ where $\mathcal{D}_i^{\mathcal{L}} \leftarrow \text{ortho}\left(\mathcal{P}_k(\nabla f(\mathbf{X}_i))\right)$
5: $\mathcal{S}_i^{\mathcal{M}} \leftarrow \mathcal{D}_i^{\mathcal{M}} \cup \mathcal{M}_i$ where $\mathcal{D}_i^{\mathcal{M}} \leftarrow \text{supp}\left(\mathcal{P}_{\Sigma_s}(\nabla f(\mathbf{X}_i))\right)$
6: **Low rank matrix estimation:**
7: $\mathbf{V}_i^{\mathcal{L}} \leftarrow \arg\min_{\mathbf{V}:\mathbf{V}\in\text{span}(\mathcal{S}_i^{\mathcal{L}})} \left\|\boldsymbol{y} - \boldsymbol{\mathcal{A}}(\mathbf{V} + \mathbf{M}_i)\right\|_2^2$
8: $\mathbf{L}_{i+1} \leftarrow \mathcal{P}_k(\mathbf{V}_i^{\mathcal{L}})$ with $\mathcal{L}_{i+1} \leftarrow \text{ortho}\,(\mathbf{L}_{i+1})$
9: **Sparse matrix estimation:**
10: $\mathbf{V}_i^{\mathcal{M}} \leftarrow \arg\min_{\mathbf{V}:\mathbf{V}\in\text{supp}(\mathcal{S}_i^{\mathcal{M}})} \left\|\boldsymbol{y} - \boldsymbol{\mathcal{A}}(\mathbf{V} + \mathbf{L}_i)\right\|_2^2$
11: $\mathbf{M}_{i+1} \leftarrow \mathcal{P}_{\Sigma_s}(\mathbf{V}_i^{\mathcal{M}})$ with $\mathcal{M}_{i+1} \leftarrow \text{supp}\,(\mathbf{M}_i)$
12: $\mathbf{X}_{i+1} \leftarrow \mathbf{L}_{i+1} + \mathbf{M}_{i+1}$
13: $i \leftarrow i + 1$
14: **until** $\|\mathbf{X}_i - \mathbf{X}_{i-1}\|_2 \leq \eta\|\mathbf{X}_i\|_2$ or MaxIterations.

Algorithm 1: SpaRCS

Notation: We reserve lower-case letters for scalar variable representation. Bold upper-case letters denote matrices while bold calligraphic upper-case letters represent linear maps. We reserve plain calligraphic upper-case letters for set representations. We denote a set of orthonormal, rank-1 matrices that span the subspace induced by $\mathbf{X}$ as $\text{ortho}(\mathbf{X})$. Given a matrix $\mathbf{X}$ and a subspace set $\mathcal{S}$ such that $\text{span}(\mathcal{S}) \subseteq \text{span}(\text{ortho}(\mathbf{X}))$, the orthogonal projection of $\mathbf{X}$ onto the subspace spanned by $\mathcal{S}$ is given by $\mathcal{P}_{\mathcal{S}}\mathbf{X}$ while $\mathcal{P}_{\mathcal{S}}^{\perp}\mathbf{X}$ represents the projection onto the subspace, orthogonal to $\text{span}(\mathcal{S})$. Given a matrix $\mathbf{X}$ and an index set $\mathcal{U}$, $(\mathbf{X})_{\mathcal{U}}$ denotes the (sub)matrix of $\mathbf{X}$ with entries in $\mathcal{U}$ while $(\mathbf{X})_{\mathcal{U}^c}$ denotes the (sub)matrix of $\mathbf{X}$ with entries in the complement set of $\mathcal{U}$. The best s-sparse and rank-k approximations of a matrix $\mathbf{X}$ are given by $\mathcal{P}_{\Sigma_s}(\mathbf{X})$ and $\mathcal{P}_k(\mathbf{X})$, respectively. For any two subspace sets $\mathcal{S}_1$, $\mathcal{S}_2$, we use the shorthand $\mathcal{P}_{\mathcal{S}_1\setminus\mathcal{S}_2}$ to denote the projection onto the subspace defined by $\mathcal{S}_1$, orthogonal to the subspace defined by $\mathcal{S}_2$—similar notation is used for index sets. We use $\mathbf{X}_i \in \mathcal{R}^{m\times n}$ to represent the current matrix estimate at the i-th iteration. The rank of $\mathbf{X}$ is denoted as $\text{rank}(\mathbf{X}) \leq \min\{m, n\}$ while the non-zero index set of $\mathbf{X}$ is given by $\text{supp}(\mathbf{X})$. The empirical data error $f(\mathbf{X}) := \|\boldsymbol{y} - \boldsymbol{\mathcal{A}}\mathbf{X}\|_2^2$ has gradient $\nabla f(\mathbf{X}) := -2\boldsymbol{\mathcal{A}}^*(\boldsymbol{y} - \boldsymbol{\mathcal{A}}\mathbf{X})$, where $\boldsymbol{\mathcal{A}}^*$ is the adjoint linear operator. $\mathbb{I}$ represents the identity matrix.

2. THE SPARCS ALGORITHM

Explicit description of SpaRCS [5] is provided in Algorithm 1 in pseudocode form. This approach borrows from a series of vector and matrix reconstruction algorithms such as CoSaMP [6] and ADMiRA [7]. In a nutshell, this algorithm simply seeks to improve the current subspace and support set selection by iteratively collecting extended sets $\mathcal{S}_i^{\mathcal{L}}$ and $\mathcal{S}_i^{\mathcal{M}}$ with $|\mathcal{S}_i^{\mathcal{L}}| \leq 2k$ and $|\mathcal{S}_i^{\mathcal{M}}| \leq 2s$, respectively. Then, s-sparse and rank-k matrices are estimated to fit the measurements in these restricted subspace/support sets using least squares techniques.

3. IMPROVED CONVERGENCE GUARANTEES

An important ingredient for our matrix analysis is the following lemma—the proof can be found in [5].

Lemma 1. *Let $\mathcal{F}$ be a support set with $|\mathcal{F}| \leq s$ and assume $\mathbf{L} \in \mathbb{R}^{m\times n}$ is a rank-k matrix. Then, given a general linear operator $\boldsymbol{\mathcal{A}} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$ satisfying both sparse- and rank-RIP, we have:*

$$\left\|(\boldsymbol{\mathcal{A}}^*\boldsymbol{\mathcal{A}}\mathbf{L})_{\mathcal{F}}\right\|_F \lesssim \delta_{s+k}(\boldsymbol{\mathcal{A}})\left\|\mathbf{L}\right\|_F, \text{ for } \min\{m, n\} \gg s \gg k.$$

where $\delta_{s+k}(\boldsymbol{\mathcal{A}})$ denotes the RIP constant of $\boldsymbol{\mathcal{A}}$ over (disjoint) sparse index and low-rank subspace sets where the combined cardinality is less than $s + k$.

We provide improved conditions for convergence for Algorithm 1. The details of the proof will be included in an extended version of the paper. The following theorem characterizes Algorithm 1:

Theorem 1. *Given the problem configuration described in (1) and (2), assume the linear operator $\boldsymbol{\mathcal{A}}$ satisfies the sparse-RIP and rank-RIP for $\delta_{4s}(\boldsymbol{\mathcal{A}}) \leq 0.075$, $\delta_{4k}(\boldsymbol{\mathcal{A}}) \leq 0.04$ and $\delta_{2s+3k}(\boldsymbol{\mathcal{A}}) \leq 0.07$. Then, the $(i+1)$-th matrix estimate $\mathbf{X}_{i+1}$ of Algorithm 1 can be decomposed into a superposition of low-rank and sparse components as $\mathbf{X}_{i+1} = \mathbf{L}_{i+1} + \mathbf{M}_{i+1}$, satisfying the recursions:*

$$\left\|\mathbf{L}^* - \mathbf{L}_{i+1}\right\|_F \leq \rho_1^{\mathcal{L}}\left\|\mathbf{L}^* - \mathbf{L}_i\right\|_F + \rho_1^{\mathcal{M}}\left\|\mathbf{M}^* - \mathbf{M}_i\right\|_F + \gamma_1\left\|\boldsymbol{\varepsilon}\right\|_2$$

$$\left\|\mathbf{M}^* - \mathbf{M}_{i+1}\right\|_F \leq \rho_2^{\mathcal{L}}\left\|\mathbf{L}^* - \mathbf{L}_i\right\|_F + \rho_2^{\mathcal{M}}\left\|\mathbf{M}^* - \mathbf{M}_i\right\|_F + \gamma_2\left\|\boldsymbol{\varepsilon}\right\|_2$$

where $\rho_1^{\mathcal{L}} = 0.1605$, $\rho_2^{\mathcal{L}} = 0.3431$, $\rho_1^{\mathcal{M}} = 0.3376$, $\rho_2^{\mathcal{M}} = 0.1414$, $\gamma_1 = 4.36$ and, $\gamma_2 = 4.45$.

To compare with state-of-the-art approaches, [5] provides the following constants for the same RIP assumptions: $\rho_1^{\mathcal{L}} = 0.479$, $\rho_2^{\mathcal{L}} = 0.474$, $\rho_1^{\mathcal{M}} = 0.47$, $\rho_2^{\mathcal{M}} = 0.324$, $\gamma_1 = 6.68$ and, $\gamma_2 = 6.88$.

Next, we sketch the proof of Theorem 1 in a modular fashion and use key ingredients to analyze our MATRIX ALPS algorithm.

3.1. Subspace and support exploration

Lemma 2 (Active subspace expansion)**.** *At each iteration, the Active Subspace Expansion step (Step 4) captures information contained in the true matrix $\mathbf{L}^*$ with $\mathcal{L}^* \leftarrow \text{ortho}(\mathbf{L}^*)$, such that:*

$$\left\|\mathcal{P}_{\mathcal{L}^*\setminus\mathcal{S}_i^{\mathcal{L}}}(\mathbf{L}^* - \mathbf{L}_i)\right\|_F \leq (2\delta_{2k}(\boldsymbol{\mathcal{A}}) + 2\delta_{3k}(\boldsymbol{\mathcal{A}}))\left\|\mathbf{L}^* - \mathbf{L}_i\right\|_F + 2\delta_{2k+2s}(\boldsymbol{\mathcal{A}})\left\|\mathbf{M}^* - \mathbf{M}_i\right\|_F + \sqrt{2(1+\delta_{2k}(\boldsymbol{\mathcal{A}}))}\left\|\boldsymbol{\varepsilon}\right\|_2.$$

Lemma 2 states that, at each iteration, the Active subspace expansion step identifies a $2k$ rank subspace in $\mathbb{R}^{m\times n}$ such that the amount of unrecovered energy of $\mathbf{L}^*$—i.e., the projection of $\mathbf{L}^*$ onto the orthogonal subspace of $\text{span}(\mathcal{S}_i^{\mathcal{L}})$—is bounded as shown above. Similarly, the next Corollary holds for the sparse estimation part:

Corollary 1 (Active support expansion)**.** *At each iteration, the Active Support Expansion step (Step 5) captures information contained in the true matrix $\mathbf{M}^*$ with $\mathcal{M}^* \leftarrow \text{supp}(\mathbf{M}^*)$, such that:*

$$\left\|\left(\mathbf{M}^* - \mathbf{M}_i\right)_{\mathcal{M}^*\setminus\mathcal{S}_i^{\mathcal{M}}}\right\|_F \leq (\delta_{2s}(\boldsymbol{\mathcal{A}}) + \delta_{4s}(\boldsymbol{\mathcal{A}}))\left\|\mathbf{M}^* - \mathbf{M}_i\right\|_F + \left(\delta_{2k+s}(\boldsymbol{\mathcal{A}}) + \delta_{2k+2s}(\boldsymbol{\mathcal{A}})\right)\left\|\mathbf{L}^* - \mathbf{L}_i\right\|_F + \sqrt{2(1+\delta_{4s}(\boldsymbol{\mathcal{A}}))}\left\|\boldsymbol{\varepsilon}\right\|_2.$$

3.2. Least-squares estimates over low rank subspaces

Lemma 3 (Least-squares error norm reduction over a low-rank subspace)**.** *Let $\mathcal{S}_i^{\mathcal{L}}$ be a set of orthonormal, rank-1 matrices such that $\text{span}\,(\mathcal{S}_i^{\mathcal{L}}) \leq 2k$. Then, the rank-$2k$ solution $\mathbf{V}_i^{\mathcal{L}}$ in Step 7 identifies most of the energy of $\mathcal{L}^*$ over $\mathcal{S}_i^{\mathcal{L}}$ such that:*

$$\left\|\mathbf{V}_i^{\mathcal{L}} - \mathbf{L}^*\right\|_F \leq \frac{1}{\sqrt{1-\delta_{3k}^2(\boldsymbol{\mathcal{A}})}}\left\|\mathcal{P}_{\mathcal{S}_i^{\mathcal{L}}}^{\perp}(\mathbf{V}_i^{\mathcal{L}} - \mathbf{L}^*)\right\|_F + \frac{(1+2\delta_{2k}(\boldsymbol{\mathcal{A}}))}{1-\delta_{3k}^2(\boldsymbol{\mathcal{A}})}\left(\delta_{2k+2s}(\boldsymbol{\mathcal{A}})\left\|\mathbf{M}^* - \mathbf{M}_i\right\|_F + \sqrt{1+\delta_{2k}(\boldsymbol{\mathcal{A}})}\left\|\boldsymbol{\varepsilon}\right\|_2\right).$$

Assuming $\boldsymbol{\mathcal{A}}$ is well-conditioned over low-rank subspaces, the main complexity of this operation is dominated by the solution of a symmetric linear system of equations. Using Lemma 3 and the following inequality:

$$\|\mathbf{L}_{i+1} - \mathbf{V}_i^{\mathcal{L}}\|_F \leq \left\|\mathcal{P}_{\mathcal{S}_i^{\mathcal{L}}}(\mathbf{V}_i^{\mathcal{L}} - \mathbf{L}^*)\right\|_F \leq \left\|\mathbf{V}_i^{\mathcal{L}} - \mathbf{L}^*\right\|_F,$$

1: **Input:** $\boldsymbol{y}$, $\boldsymbol{\mathcal{A}}$, $\boldsymbol{\mathcal{A}}^*$, Tolerance η, MaxIterations, $\tau_i,\ \forall i$
2: **Initialize:** $\{\mathbf{Q}_0, \mathbf{M}_0, \mathbf{L}_0\} \leftarrow 0$, $\{\mathcal{L}_0, \mathcal{M}_0\} \leftarrow \{\emptyset\}$, $i \leftarrow 0$
3: **repeat**
4: **Low rank matrix estimation:**
5: $\mathcal{D}_i^{\mathcal{L}} \leftarrow \text{ortho}\left(\mathcal{P}_k(\nabla f(\mathbf{Q}_i))\right)$
6: $\mathcal{S}_i^{\mathcal{L}} \leftarrow \mathcal{D}_i^{\mathcal{L}} \cup \mathcal{L}_i$
7: $\mathbf{V}_i^{\mathcal{L}} \leftarrow \mathbf{Q}_i^{\mathcal{L}} - \frac{\mu_i^{\mathcal{L}}}{2}\mathcal{P}_{\mathcal{S}_i^{\mathcal{L}}}\nabla f(\mathbf{Q}_i)$
8: $\mathbf{L}_{i+1} \leftarrow \mathcal{P}_k(\mathbf{V}_i^{\mathcal{L}})$ with $\mathcal{L}_{i+1} \leftarrow \text{ortho}\,(\mathbf{L}_{i+1})$
9: $\mathbf{Q}_{i+1}^{\mathcal{L}} \leftarrow \mathbf{L}_{i+1} + \tau_i(\mathbf{L}_{i+1} - \mathbf{L}_i)$
10: $\mathbf{Q}_{i+1} \leftarrow \mathbf{Q}_{i+1}^{\mathcal{L}} + \mathbf{Q}_i^{\mathcal{M}}$
11: **Sparse matrix estimation:**
12: $\mathcal{D}_i^{\mathcal{M}} \leftarrow \text{supp}\left(\mathcal{P}_{\Sigma_s}(\nabla f(\mathbf{Q}_{i+1}))\right)$
13: $\mathcal{S}_i^{\mathcal{M}} \leftarrow \mathcal{D}_i^{\mathcal{M}} \cup \mathcal{M}_i$
14: $(\mathbf{V}_i^{\mathcal{M}})_{\mathcal{S}_i^{\mathcal{M}}} \leftarrow (\mathbf{Q}_i^{\mathcal{M}})_{\mathcal{S}_i^{\mathcal{M}}} - \frac{\mu_i^{\mathcal{M}}}{2}(\nabla f(\mathbf{Q}_{i+1}))_{\mathcal{S}_i^{\mathcal{M}}}$
15: $\mathbf{M}_{i+1} \leftarrow \mathcal{P}_{\Sigma_s}(\mathbf{V}_i^{\mathcal{M}})$ with $\mathcal{M}_{i+1} \leftarrow \text{supp}\,(\mathbf{M}_{i+1})$
16: $\mathbf{Q}_{i+1}^{\mathcal{M}} \leftarrow \mathbf{M}_{i+1} + \tau_i(\mathbf{M}_{i+1} - \mathbf{M}_i)$
17: $\mathbf{Q}_{i+1} \leftarrow \mathbf{Q}_{i+1}^{\mathcal{L}} + \mathbf{Q}_{i+1}^{\mathcal{M}}$
18: $i \leftarrow i + 1$
19: **until** $\|\mathbf{X}_i - \mathbf{X}_{i-1}\|_2 \leq \eta\|\mathbf{X}_i\|_2$ or MaxIterations.

Algorithm 2: MATRIX ALPS Instance

which is due to the best rank-k subspace selection on $\mathbf{V}_i^{\mathcal{L}}$ (Step 8), the following inequality holds true:

$$\left\|\mathbf{L}_{i+1} - \mathbf{L}^*\right\|_F \leq \sqrt{\frac{1+3\delta_{3k}^2(\boldsymbol{\mathcal{A}})}{1-\delta_{3k}^2(\boldsymbol{\mathcal{A}})}}\left\|\mathcal{P}_{\mathcal{S}_i^{\mathcal{L}}}^{\perp}(\mathbf{V}_i^{\mathcal{L}} - \mathbf{L}^*)\right\|_F +$$
$$\left(\sqrt{1+3\delta_{3k}^2(\boldsymbol{\mathcal{A}})}\cdot\frac{1+2\delta_{2k}(\boldsymbol{\mathcal{A}})}{1-\delta_{3k}^2(\boldsymbol{\mathcal{A}})} + \sqrt{3}\right)\left(\delta_{2s+2k}(\boldsymbol{\mathcal{A}})\left\|\mathbf{M}^* - \mathbf{M}_i\right\|_F\right.$$
$$\left. + \sqrt{1+\delta_{2s}(\boldsymbol{\mathcal{A}})}\left\|\boldsymbol{\varepsilon}\right\|_2\right). \quad (7)$$

Combining Lemma 2 with the inequality (7), we obtain the first inequality in Theorem 1.

3.3. Least-squares estimates over sparse support sets

Using similar techniques descibed above for the sparse matrix estimate, we derive the following result:

Corollary 2 (Least-squares error norm reduction over sparse support sets)**.** *Let $\mathcal{S}_i^{\mathcal{M}} \subseteq \{(i,j) : i \in \{1,\ldots,m\},\ j \in \{1,\ldots n\}\}$ be a $2s$-sparse index set. Then, the $2s$-sparse matrix $\mathbf{V}_i^{\mathcal{M}}$ (Step 10) identifies energy of $\mathbf{M}^*$ over $\mathcal{S}_i^{\mathcal{M}}$ such that:*

$$\left\|\mathbf{V}_i^{\mathcal{M}} - \mathbf{M}^*\right\|_F \leq \frac{1}{\sqrt{1-\delta_{4s}^2(\boldsymbol{\mathcal{A}})}}\left\|\left(\mathbf{V}_i^{\mathcal{M}} - \mathbf{M}^*\right)_{(\mathcal{S}_i^{\mathcal{M}})^c}\right\|_F +$$
$$\frac{(1+2\delta_{2s}(\boldsymbol{\mathcal{A}}))}{1-\delta_{4s}^2(\boldsymbol{\mathcal{A}})}\left(\delta_{3s+2k}(\boldsymbol{\mathcal{A}})\left\|\mathbf{L}^* - \mathbf{L}_i\right\|_F + \sqrt{1+\delta_{3s}(\boldsymbol{\mathcal{A}})}\left\|\boldsymbol{\varepsilon}\right\|_2\right).$$

In sequence, we follow the same motions to obtain an inequality analogous to (7) for the sparse matrix estimate part.

4. THE MATRIX ALPS FRAMEWORK

To accelerate the convergence speed of SpaRCS, we propose MATRIX ALPS algorithm based on acceleration techniques from convex analysis [11, 12]. At each iteration, we leverage both low rank and sparse matrix estimates from previous iterations to form a gradient surrogate with low-computational cost. Then, we update the current estimates using memory to gain momentum in convergence as proposed in Nesterov's optimal gradient methods. A key ingredient is the selection of the momentum term τ—constant and adaptive momentum selection strategies can be found in [12]. We reserve the analysis for the adaptive case for an extended paper.

To further improve the convergence speed, we replace the least-squares optimization steps with first-order gradient descent updates—the step size $\mu_i^{\mathcal{L}}$, $\mu_i^{\mathcal{M}}$ selections follow from [12].

The best projection of an arbitrary matrix onto the set of low rank matrices requires sophisticated matrix decompositions such as Singular Value Decomposition (SVD). Using the Lanczos approach, we require $O(kmn)$ arithmetic operations to compute a rank-k matrix approximation for a given constant accuracy—a prohibitive time-complexity that does not scale well for many practical applications. Alternatives to SVD can be found in [4,13]. Furthermore, [14] includes ϵ-approximate low rank matrix projections in the recovery process and study their effects on the convergence.

The following theorem characterizes Algorithm 2 for the noiseless case using a constant momentum step size selection strategy.

Theorem 2. *Let $\boldsymbol{\mathcal{A}} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$ be a linear operator satisfying rank-RIP and sparse-RIP with constants $\delta_{4k}(\boldsymbol{\mathcal{A}}) \leq 0.09$ and $\delta_{4s}(\boldsymbol{\mathcal{A}}) \leq 0.095$, respectively. Furthermore, assume constant momentum step size selection with $\tau_i = 1/4,\ \forall i$. We consider the noiseless case where the set of observations satisfy $\boldsymbol{y} = \boldsymbol{\mathcal{A}}\mathbf{X}^*$ for $\mathbf{X}^* := \mathbf{L}^* + \mathbf{M}^*$ as defined in* PROBLEM*. Then, Algorithm 2 satisfies the following second-order linear system:*

$$\mathbf{x}(i+1) \leq (1+\tau)\boldsymbol{\Delta}\mathbf{x}(i) + \tau\boldsymbol{\Delta}\mathbf{x}(i-1), \quad (8)$$

where $\mathbf{x}(i) := \begin{bmatrix}\left\|\mathbf{L}_i - \mathbf{L}^*\right\|_F \\ \left\|\mathbf{M}_i - \mathbf{M}^*\right\|_F\end{bmatrix}$ and $\boldsymbol{\Delta} := \begin{bmatrix}\Delta_{11} & \Delta_{12} \\ \Delta_{21} & \Delta_{22}\end{bmatrix}$ depends on RIP constants $\delta_{4k}(\boldsymbol{\mathcal{A}})$ and $\delta_{4s}(\boldsymbol{\mathcal{A}})$. Furthermore, the above inequality can be transformed into the following first-order linear system:

$$\mathbf{w}(i+1) \leq \underbrace{\begin{bmatrix}(1+\tau)\boldsymbol{\Delta} & \tau\boldsymbol{\Delta} \\ \mathbb{I} & \mathbf{0}\end{bmatrix}^i}_{\widehat{\boldsymbol{\Delta}}}\mathbf{w}(0), \quad (9)$$

for $\mathbf{w}(i) := [\mathbf{x}(i+1)\ \ \mathbf{x}(i)]^T$. We observe that $\lim_{i\to\infty}\mathbf{w}(i) = \mathbf{0}$ since $|\lambda_j(\widehat{\boldsymbol{\Delta}})| \leq 1,\ \forall j$.

Due to space constraints, we reserve the proof as well as the noisy analog of Theorem 2 for an extended version of the paper.

5. EXPERIMENTS

Robust matrix completion:[1] The rank-k $\mathbf{X}^* \in \mathbb{R}^{m\times n}$ is synthesized as $\mathbf{X}^* := \mathbf{U}\mathbf{R}^T$ where $\mathbf{U} \in \mathbb{R}^{m\times k}$ and $\mathbf{R} \in \mathbb{R}^{n\times k}$ and $\left\|\mathbf{X}^*\right\|_F = 1$. We subsample $\mathbf{X}^*$ by observing $p = 0.3mn$ entries, drawn uniformly at random. The set of observations satisfies: $\boldsymbol{y} = \boldsymbol{\mathcal{A}}_\Omega\mathbf{X}^* + \boldsymbol{\varepsilon}$. Here, Ω denotes the set of ordered pairs that represent the coordinates of the observable entries and $\boldsymbol{\mathcal{A}}_\Omega$ denotes the linear operator (mask) that subsamples a matrix according to Ω.

We generate various problem configurations, both for noisy and noiseless settings. All the algorithms are tested for the same signal-matrix-noise realizations and use the same tolerance parameter $\eta = 10^{-4}$. For fairness, we modified all the algorithms so that *they exploit the true rank*. For low-rank projections, we use PROPACK package [15], except [9] which is SVD-less. We changed the maximum number of cycles in [9] from 150 to 30 to improve its speed. A summary of the results can be found in Fig. 1. We observe that MATRIX ALPS has better phase transition performance over various k. A complete comparison using randomized, low-rank projection schemes in MATRIX ALPS is provided in the extended paper.

[1] Codes are available for MATLAB at *http://lions.epfl.ch/MatrixALPS*

$m \times n$	k	$\|\boldsymbol{\varepsilon}\|_2$	Iterations	Relative Error := $\|\widehat{\mathbf{X}} - \mathbf{X}^*\|_F / \|\mathbf{X}^*\|_F$ (10^{-3})	Time (sec)
200×400	5	0	$29\ /\ 24/\ -\ /46/11$	$0.134/0.18/0.002/0.78/0.04$	$2.26/0.27/0.95/0.36/\mathbf{0.21}$
200×400	5	10^{-2}	$29/24/\ -\ /45/11$	$0.127/0.164/0.01/0.76/0.05$	$2.16/0.26/0.96/0.36/\mathbf{0.23}$
200×400	10	10^{-2}	$700/33/\ -\ /63/15$	$6.7/0.5/0.01/1.2/0.1$	$36.38/0.45/1.13/0.64/\mathbf{0.37}$
200×400	15	0	$700/48/\ -\ /88/22$	$150/0.93/340/2.1/0.15$	$98.12/0.82/1.29/1.08/\mathbf{0.68}$
1000×5000	10	0	$-/22/\ -\ /30/6$	$-/0.09/0.008/0.34/0.03$	$-/10.8/27.6/10.2/\mathbf{5.5}$
1000×5000	50	10^{-4}	$-/24/\ -\ /48/10$	$-/0.2/0.002/0.73/0.11$	$-/23.4/171.37/35.5/\mathbf{17.2}$
1000×5000	120	0	$-/63/\ -\ /118/26$	$-/0.52/0.07/1.22/0.077$	$-/139/501/228/\mathbf{101}$

Fig. 1. Comparison table for the matrix completion problem. Table depicts median values over 50 Monte-Carlo iterations. To separate the results, we use "/". The list of algorithms includes: SpaRCS [5] / ALM [8] / GROUSE [9] / SVP [10] / MATRIX ALPS. Bold numbers highlight the fastest convergenvce in execution time. "−" denotes either no information or not applicable due to slow convergence.

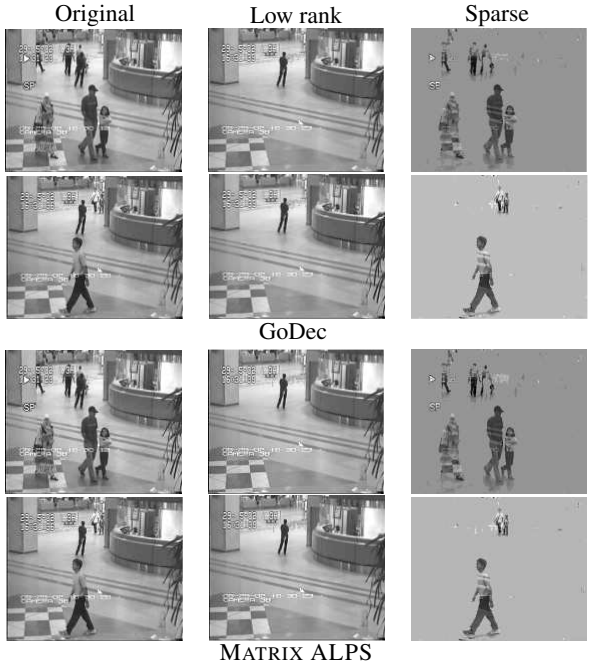

Fig. 2. Background subtraction in video sequence. Median execution times over 10 Monte-Carlo iterations. GoDec: 34.8 sec—MATRIX ALPS: 15.8 sec.

RPCA: We consider the problem of background subtraction in video sequences: static brackground scenes are considered low-rank while moving foreground objects are sparse data. Using the complete set of measurements, this problem falls under the RPCA framework. We apply the GoDec algorithm [4] and the MATRIX ALPS scheme on a 144 x 176 x 200 video sequence. Both solvers use the same low-rank projection operators based on randomized QR factorization ideas [4, 13]. Representative results are depicted in Fig. 2.

6. CONCLUSIONS

We study the general problem of sparse plus low rank matrix recovery from incomplete and noisy data. In essence, the problem under consideration includes various low-dimensional models as special cases such as sparse signal reconstruction, affine rank minimization and robust PCA. Based on this algorithm, we derive improved conditions on the restricted isometry constants that guarantee successful reconstruction. Furthermore, we show that the memory-based scheme provides great computational advantage over both the convex and the non-convex approaches.

7. REFERENCES

[1] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory*, 2006.

[2] B. Recht M. Fazel and P. A. Parrilo. Guaranteed minimum rank solutions to linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.

[3] E.J. Candes, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Arxiv preprint ArXiv:0912.3599*, 2009.

[4] T. Zhou and D. Tao. Godec: Randomized low-rank & sparse matrix decomposition in noisy case. In *ICML*, 2011.

[5] A.E. Waters, A.C. Sankaranarayanan, and R.G. Baraniuk. Sparcs: Recovering low-rank and sparse matrices from compressive measurements. *NIPS*, 2011.

[6] D. Needell and J. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, June 2008.

[7] K. Lee and Y. Bresler. Admira: Atomic decomposition for minimum rank approximation. *IEEE Trans. Inf. Theory*, 2010.

[8] L. Wu Z. Lin, M. Chen and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *UIUC Technical Report UILU-ENG-09-2215*.

[9] L. Balzano, R. Nowak, and B. Recht. Online identification and tracking of subspaces from highly incomplete information. In *Annual Allerton Conference*. IEEE, 2010.

[10] R. Meka, P. Jain, and I. S. Dhillon. Guaranteed rank minimization via singular value projection. In *NIPS*, 2010.

[11] Y. Nesterov. *Introductory lectures on convex optimization*. Kluwer Academic Publishers, 1996.

[12] A. Kyrillidis and V. Cevher. Recipes on hard thresholding methods. *CAMSAP*, 2011.

[13] N. Halko, P.G. Martinsson, and J.A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *Arxiv preprint arXiv:0909.4061*, 2009.

[14] A. Kyrillidis and V. Cevher. Matrix algebraic pursuits: Recipes for hard thresholding methods. *Technical Report*, 2012.

[15] R. M. Larsen. Propack: Software for large and sparse svd calculations. http://soi.stanford.edu/rmunk/PROPACK.